```
2           000000          CRBINT=0
3           000001          RSX=1
```

```
                                ;PREFIX FILE FOR RSX11M ASSEMBLER AND ANY OTHER ASSEMBLER
                                ;WHICH DOES NOT RECOGNIZE OPCODE MNEMONICS EXCEPT WHEN THEY
                                ;APPEAR IN THE OPCODE FIELD
000000                          HALT=0
000240                          NOP=240
000241                          CLC=241
000242                          CLV=242
000244                          CLZ=244
000250                          CLN=250
000257                          CCC=257
000261                          SEC=261
000262                          SEV=262
000264                          SEZ=264
000270                          SEN=270
000277                          SCC=277
```

```
                        ;COMTEX-11 BISYNC TAP
                        ;
                        ; COPYRIGHT (C)  1975, 1976, 1977
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                        ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                        ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                        ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                        ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                        ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                        ;
                        ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ; DEC.

                                .TITLE BISYNC
                                .IDENT /V00237/
                                ;TAPE 1 OF 5

                                ;REVISION HISTORY
                                ;
                                ; NOV 1977 - CEP (EDIT 41)
                                ; CHANGED SETEOF CODE TO IGNORE AN EOT IN RESPONSE TO BID
                                ;
                                ; NOV 1977 - CEP (EDIT 40)
                                ; PUT IN TIMEOUT COUNTERS
                                ;
                                ; SEP 1977 - CEP (EDIT 37)
                                ; CODE DELETED FOR CUT CODE CHANGES
                                ; JUN 1977 - CEP (EDIT 36)
                                ;
                                ;CHANGES TO ALLOW PACKAGE TO RUN ON AN LSI11
                                ;REQUIRES MODIFICATION TO THE HANDLING OF THE PS AND KG11
                                ; MAR 1977 - CEP (EDIT 35)
                                ;
                                ; CHANGE KG11 CODE SINCE IT IS SINGLE LINE - RSTS ONLY
                                ;
                                ;MAR 1977 - CEP (EDIT 34)
                                ;
                                ;PUT IN NAK AND TIMEOUT COUNTS FOR USE BY RSTS ONLY
                                ;
                                ;
                                ; FEB 1977 - CEP  (EDIT 33)
                                ;
                                ; MOVED RESETTING OF USER OVERFLOW BIT TO AVOID A
                                ; TIMING ERROR WHICH LOST THE SECOND RECOPD
                                ;
                                ; JAN 1977 - CEP (EDIT 32)
                                ;
                                ; PUT IN CONDITIONAL ASSEMBLIES TO CUT BACK SIZE OF CODE
```

```
                        ;
                        ; SEPT 1976 - CEP (EDIT 31)
                        ;
                        ; PUT IN SPECIAL CHECK FOR PERMANENT NATIVE MODE POSSIBLE IN
                        ; RSTS - SOLVES TIMING PROBLEM
                        ;
                        ; SEPT 1976 - CEP (EDIT 30)
                        ;
                        ; PUT IN CHECK FOR BLOCK SIZE MINUS 6 ON ITB - FIXES A
                        ; MULTIPLE RECORD TRANSMISSION IN GEN MODE
                        ;
                        ; SEPT 1976 - CEP (EDIT 27)
                        ;
                        ; SET A FLAG WHEN BEING 'WAK'ED SO THAT THE RSTS
                        ; 30 SECOND TIMER CAN TELL WE ARE IN WAKS AND NOT HUNG
                        ;
                        ;
                        ; AUGUST 15, 1975 - RHR (EDIT 26)
                        ;
                        ; REINSTALL ACK AVAILABLE FLAG IN THE LINE STATUS WORD.
                        ; IT IS NEEDED TO INFORM THE TAP AS TO WHEN THE UAP IS FINISHED
                        ; PROCESSING A BLOCK OF DATA.  ALSO, DELETE THE SEND ACK
                        ; FLAG IN THE LINE STATUS WORD (IT IS NOT USED!?)
                        ;
                        ; JULY 16, 1975 - RHR (EDIT 25)
                        ;
                        ; FIX PUTTC CODE TO ALLOW A PUTTC TO CHANGE MODES IN THE MIDDLE
                        ; OF A MESSAGE TRANSFER. THIS WILL PERMIT THE SENDING OF NATIVE
                        ; AND NON-NATIVE DATA IN THE SAME TRANSMISSION.
                        ;
                        ; JUNE 23, 1975 - RHR (EDIT 24)
                        ;
                        ; REVAMP THE TIMER CODE.  ESTABLISH A BYTE IN
                        ; THE LINE TABLE (SRC & MRC) WHICH CONTAINS A COUNTER.
                        ; THIS COUNTER WILL BE DECREMENTED EVERY TIME A 1/2 SECOND TIMER
                        ; GOES OFF.  IF THE COUNTER IS ZERO THE TIMEOUT ROUTINE FOR
                        ; THE TIMER WILL BE CALLED.  THE INTERVAL TIMER
                        ; IS STARTED AT INITIALIZATION TIME AND IS ALWAYS TICKING
                        ; THIS WILL SOLVE THE TIMER QUEUE OVERFLOW PROBLEM WITH RSX
                        ; AND STILL BE ABLE TO IMPLEMENT THE REAL SLAVE RECV. TIMER.
                        ; N.B.  THIS EDIT DELETES THE EDIT OF 7/9/74 CONCERNING THE TIMER.
                        ;
                        ; JUNE 17, 1975 - RHR (EDIT 23)
                        ;
                        ; MODIFY INITIALIZATION CALL PARAMETER LIST IF UAPTIM
                        ; IS DEFINED SO THAT USER PROGRAM SUPPLIES ALL TIMER VALUES
                        ; THIS WILL ALLOW THE UAP TO TAILOR THE TIMERS TO BE DEPENDENT
                        ; ON LINE SPEED AND ON TICS PER SECOND.  THE IMPLEMENTATION
                        ; ASSUMES THE SAME VALUE FOR ALL LINES (NOTE: AS OF THIS
                        ; EDIT THE TAP WILL NOT RUN MULTIPLE LINES).
                        ;
                        ; JUNE 16, 1975 - RHR (EDIT 22)
                        ;
                        ; FIX WACK HANDLING CODE TO LET EITHER THE ENQ OR MASTER TIMER
                        ; CAUSE THE NEXT ENQ TO BE SENT.  THIS WILL PREVENT TOO MANY TIMERS
                        ; GETTING QUEUED WITH THE RSX-11 VERSION.  N.B.  THE MTOUT BIT
                        ; IN THE TRANSMIT STATUS WORD NOW INDICATES TIMEOUT
```

```
                                        ; OR WACK RECEIVED.
                                        ;
                                        ; MAY 28, 1975 - RHR (EDIT 21)
                                        ;
                                        ; ADD TRANSPARENT RECORD INDICATION TO RECEIVE COMPLETE
                                        ;STATUS RETURN (TRNREC IN RSW).
                                        ;
                                        ; APRIL 28, 1975 - RHR (EDIT 20)
                                        ;
                                        ;GLOBALIZE THE EBCDIC TO ASCII TRANSLATION TABLE FOR THE UAP
                                        ;EXPAND TRANSMIT DONE UIB ENTRY TO 4 WORDS WHERE THE FOURTH
                                        ;WORD IS THE NUMBER OF RECEIVED NAKS FOR THIS BLOCK
                                        ;
                                        ; FEBRUARY 12, 1975 - RHR (EDIT 17)
                                        ;
                                        ;INSERT TTD SEQUENCE DETECTION.  TAP SHOULD, UPON FINDING
                                        ;AN ENQ FOLLOWING AN STX IN TEXT, SET A NAK EXPECTED FLAG
                                        ;AND TRANSMIT THE BUFFER IMMEDIATELY.  UPON RECEIVING
                                        ;THE NAK IT WILL GENERATE A UIB ENTERY
                                        ;
                                        ; JANUARY 31, 1975 - RHR (EDIT 16)
                                        ;
                                        ;REMOVE LAST REMENANTS OF LINE TURNAROUND DELAY - ACK AVAILABLE
                                        ;FLAG
                                        ;
                                        ; JANUARY 30, 1975 - RHR
                                        ;
                                        ;FIX SO THAT MASTER TIMEOUT FLAG DOES NOT GET CLEARED
                                        ;IN TRANSMIT COMPLETE CODE.
                                        ;ALSO, FIX A RSTS STACK POPPING PROBLEM IN THE PUTTC CODE
                                        ;
                                        ; JANUARY 23, 1975 - RHR
                                        ;
                                        ;CHANGE DEFINITION OF "THALT", TIMER QUEUE OVERFLOW HALT,
                                        ;TO A "NOP" SO RSTS AND UNMAPPED RSX11M DON'T HALT BUT A
                                        ;WORD IS STILL SAVED SO IT CAN BE PATCHED FOR DIAGNOSTIC PURPOSES
                                        ;
                                        ; DECEMBER 27, 1974 - RHR
                                        ;
                                        ;DON'T ALLOW ANOTHER SLAVE RECEIVE TIMER TO BE STARTED IF
                                        ; ONE ALREADY IS GOING.  THIS FIXES A PROBLEM FOUND WHEN
                                        ; RUNNING IN TRANSPARENT MODE IN WHICH A TIMER WOULD GET
                                        ; QUEUED ON EACH "DLE-STX" SEQUENCE OF THE MESSAGE.  UNDER
                                        ; CERTAIN CONDITIONS THE TIMER QUEUE WOULD OVERFLOW WITH
                                        ; UNNECESARY TIMERS.
                                        ;
                                        ; DECEMBER 5,1974 - RHR
                                        ;
                                        ;CORRECT REVISION OF 8/13/74
                                        ;REQUEST A RESYNC AFTER WRONG ACK IS RECEIVED IN CASE THE
                                        ;LINE IS A FULL DUPLEX LEASED LINE
                                        ;
                                        ;AUGUST 16, 1974 - RHR
                                        ;
                                        ;CLEAN UP CODE A BIT
                                        ;
                                        ; AUGUST 13, 1974 - RHR
```

```
172                                     ;
173                                     ;CHECK FOR WRONG ACK AFTER AN ENQ REQUEST FOR SLAVES LAST
174                                     ; RESPONSE, I.E. RECEIVING STATION NEVER
175                                     ; RECEIVED THE LAST BLOCK WE SENT, WE TIMED OUT
176                                     ; AND REQUESTED A RESPONSE AND SENT US BACK THE ACK FOR
177                                     ; THE PREVIOUS BLOCK
178                                     ;
179                                     ;
180                                     ;JULY 9, 1974 - RHR
181                                     ;
182                                     ;ADD CONDITIONAL ASSEMBLY PARAMETER "RSX11M". DEFINE FOR
183                                     ; USE WITH RSX11M. (MAINLY USED TO CAUSE THE M-ASSEMBLER
184                                     ; IGNORE .EOT'S SINCE IT WILL NOT ORDINARILY DO IT)
185                                     ;JUNE 13, 1974 - RHR
186                                     ;
187                                     ;CAUSE "RVI" TO TOGGLE ACK COUNTER
188                                     ;REVAMP SLAVE RECEIVE TIMER SO THAT IT IS SET WHEN MESSAGE
189                                     ; IN PROGRESS IS SET (MIP) AND CLEARED ONLY WHEN THE MESSAGE
190                                     ; HAS BEEN COMPLETED (I.E. AT ETB OR ETX).  THE TIMER
191                                     ; LENGTH HAS ALSO BEEN EXTENDED.  THIS CHANGE WILL PREVENT
192                                     ; THE TIMER QUEUE FROM OVERFLOWING.
193                                     ; N.B.  TO USE THE STANDARD IBM3780 RECEIVE TIMER DEFINITION
194                                     ;       DEFINE THE ASSEMBLY CONDITIONAL "STDSRT"
195                                     ;IF A LEGAL FIRST CHARACTER (I.E. STX,SOH,EOT,NAK,ENQ...) IS
196                                     ; NOT RECOGNIZED AFTER THE FIRST 7 CHARACTERS
197                                     ; RECEIVED THEN FLUSH THE BUFFER AND RESET CRB TIMER.
198                                     ;ADD CONDITIONAL ASSEMBLY SWITCH ASCI TO DENOTE ASCII
199                                     ; INPUT FROM USER PROGRAM IF ASCI IS DEFINED
200                                     ;
201                                     ;JUNE 12, 1974 - RHR
202                                     ;
203                                     ;CAUSE TAP TO APPEND PAD CHARACTER IN ABNORMAL TRANSMIT
204                                     ;INSTEAD OF RECEIVING IT FROM USER PROGRAM TO AVOID TRANSLATION
205                                     ;TABLE CONFLICT BETWEEN DEL AND PAD (BOTH ASCII 177)
206                                     ;
207                                     ;JUNE 11, 1974 - RHR
208                                     ;
209                                     ;REMOVE LINE TURNAROUND DELAY TIME - HARDWARE TAKES CARE OF THIS
210                                     ;
211                                     ;JANUARY 14,1974
212                                     ;
213                                     ;3780 EMULATOR CHANGES MAY BE ASSEMBLED BY DEFINING R3780
214                                     ;
215
216                                     ;JANUARY 9, 1974
217
218                                     ;THIS REVISION PROVIDES THE CHANGES FOR THE RSX
219                                     ;AND RSTS-E VERSIONS AND ADDS CONDITIONAL ASSEMBLY
220                                     ;PARAMETER CPD26.  IF CPD26 IS NON-ZERO FEATURES OF THE
221                                     ;BISYNC TAP WHICH ARE NOT USED BY THE 2780 RCS ARE
222                                     ;NOT ASSEMBLED.
223                                     ;
224                                     ;SEPTEMBER 13,1973
225                                     ;RESET RCVE TIMER WHEN DLE ITB IS RECEIVED
226                                     ;DELETE STX FROM ITB RECORD BLOCKS (NON-TRANSPARENT)
227                                     .IF DF  RSTS
228                                     OS=0
```

```
                                ASCI=0          ;RSTS USER PROGRAM SUPPLIES ASCII CHARACTERS
                                .ENDC
                                .IF DF RT11
                                ASCI=0
                                .ENDC

                                .IF DF  RSX
000000                          UAPTIM=0                ;UAP SUPPLIES TIMER VALUES
000000                          OS=0
000000                          ASCI=0          ;RSX USER PROGRAM SUPPLIES ASCII CHARACTERS
                                .ENDC

000001                  CPD26=1
                                .IF DF LSI11
                                .GLOBL  $CRC16,CRCCHR,CRCACC
                                .ENDC

                                .GLOBL CTIME,CTDEL
                                .GLOBL  C.SARP,C.SFY
                                .GLOBL  C.BPD,C.BPTC,C.BARB,C.BINT
                                .GLOBL  C.BLTL
                                .GLOBL  ETA             ;EBCDIC TO ASCII TRANSLATION TABLE
                                .IF NDF RSTS&RSX&RT11
                                .GLOBL  CR.TBL
                                .ENDC
                                .GLOBL  C.BMSE,C.BODE

                        ;DEFINITION OF REGISTERS, ETC.
177776                          C.STAT=177776           ;ADDRESS OF PROC. STATUS REGISTER
000000                          R0=%0
000001                          R1=%1
000002                          R2=%2
000003                          R3=%3
000004                          R4=%4
000005                          R5=%5
000006                          R6=%6
000007                          R7=%7
000007                          PC=%7
000006                          SP=%6
000005                          CBAX=%5         ;CRB UNLOADER REGISTER DEFINITIONS
000004                          CBBX=%4         ;MUST NOT BE REDEFINED FOR USE WITH
000007                          RETNX=%7        ;THE DP11 ISR.

                        ; KG11 DEFINITIONS
170700                  KGSTAT=170700           ;STATUS/CONTROL REGISTER
170702                  KGBCCR=KGSTAT+2         ;BCC REGISTER
170704                  KGDATA=KGSTAT+4         ;DATA REGISTER

                        ;SPECIAL INSTRUCTION MNEMONICS
000252                  CLNV=CLN!CLV
000272                  SENV=SEN!SEV
                        ;
                        .IF NDF THALT
                        THALT=NOP               ;TIMER QUEUE OVERFLOW ACTION - PATCH TO HALT IF NECESSARY
                        .ENDC
```

```
                                ;CONDITIONAL ASSEMBLY PARAMETERS
                                ;
                                ;DEFINE THE FOLLOWING TO BE NON-ZERO TO IMPLEMENT THE FEATURE

000000                                  CA.ASC=0                ;IMPLEMENTS ASCII FACILITY(NOT IMPLEMENTED)
000001                                  CA.EBC=1                ;IMPLEMENTS EBCDIC FACILITY
000000                                  CA.SBT=0                ;IMPLEMENTS SBT FACILITY (NOT IMPLEMENTED)
000001                                  CA.NCR=1                ;DEFINE AS 0 IF BITAP LINKED WITH CR11 ISR/TAP
                                ;
                                ;DEFINE THE FOLLOWING SYMBOLS (TO ANY VALUE) TO IMPLEMENT THE
                                ;CORRESPONDING FEATURE
                                ;
                                ;RSTS -   DEFINE TO RUN WITH RSTS/E
                                ;
                                ;RSX  -   DEFINE TO RUN WITH RSX-11D/M
                                ;
                                ;RT11  -  DEFINE TO RUN WITH RT11
                                ;
                                ;ASCI -   DEFINE IF USER PROGRAM SUPPLIES ASCII DATA (AUTOMATICALLY
                                ;         DEFINED IF RSTS, RSX OR RT11 IS DEFINED)
                                ;
                                ;R3780 -  DEFINE FOR IBM 3780 EMULATOR
                                ;
                                ;OS    -   DEFINE IF RUNNING WITH AN OPERATING SYSTEM
                                ;         (AUTOMATICALLY DEFINED FOR RSX, RT11, AND RSTS)
                                ;
                                ;CDP26 -  PACKED HOLLERITH INPUT
                                ;
                                ;UAPTIM - USER PROGRAM SUPPLIES TIMER VALUES IN TICS AS FOLLOWS:
                                ;         LIST+6:  CONTENTION TIMER
                                ;         LIST+10:  # OF TICKS IN AN INTERVAL
                                ;         LIST+12:  # OF TICINT FOR MASTER TIMER
                                ;         LIST+14:  # OF TICINT FOR SLAVE RECEIVE TIMER
                                ;         LIST+16:  CRB TIMER

                                ;TIMEOUT VALUE FOR ENQ CONTENTION
                                        .IF DF  UAPTIM
000000                                  .PSECT  XXX
000000  000000                  CONTIM: .WORD   0
                                        .IFF
                                        CONTIM=60.              ;THIS IS FOR PRIMARY (1 SEC)
                                ;       CONTIM=180.             ;THIS IS FOR SECONDARY (3 SEC)
                                        .ENDC

                                ; ENQ AND NAK RETRY COUNTS
000010                                  ENQCNT=10
000010                                  NAKCNT=10
                                ; NUMBER OF BUFFERS SUPPORTED BY THE TAP (PER LINE)
000007                          C.BUFF= 7
000034                          MBX=C.BUFF+C.BUFF+C.BUFF+C.BUFF ;MAX BUFFER SLOT INDEX

                                ; SIZE OF THE TAP INTERNAL BUFFER (IN LINE TABLE)
000620                                  BSIZE=400.

                                ;TIMER VALUE FOR MASTER STATION RECEIVE TIMEOUT
                                        .IF DF  UAPTIM
000002  000000                  MSRT:   .WORD   0
```

```
                                .IFF
                                MSRT=180./TICINT                ;3 SECONDS
                                .ENDC

                            ;TIMER VALUE FOR SLAVE STATION RECEIVE TIMEOUT
                                .IF DF  UAPTIM
000004  000000              SSRT:   .WORD   0
                                .ENDC
                                .IF NDF UAPTIM
                                SSRT=180./TICINT
                                .ENDC

                                .IF DF  UAPTIM
000006  000000              TICINT: .WORD   0       ;INTERVAL FOR MASTER/SLAVE TIMERS
                                .IFF
                                TICINT=30.      ;INTERVAL FOR SLAVE RECEIVE TIMER = 1/2 SECOND
                                                ;THE ACTUAL LENGTH OF THE SLAVE RECEIVE TIMER
                                                ;  IS TICINT*SSRT
                                .ENDC
                            ;NUMBER OF LEADING GRAPHICS PERMITED BEFORE A CONTROL SEQUENCE
        000007                  LDGRPH=7.
                                .IF NDF RCVSIB
                                RCVSIB=0
                                .ENDC
                                .IF NDF CRBINT
                                CRBINT=0
                                .ENDC
```

```
372                                     ;  ASCII CHARACTER DEFINITIONS
373
374             000020                  DLE=20
375             000002                  STX=2
376             000001                  SOH=1
377             000003                  ETX=3
378             000004                  EOT=4
379             000005                  ENQ=5
380             000025                  NAK=25
381             000026                  SYN=26
382             000027                  ETB=27
383             000037                  ITB=37
384             000177                  PAD=177
385             000036                  IRS=36
386
```

```
                                ; LINE TABLE DEFINITIONS
000002                          IPL=2                   ;ISR PARAMETER LIST
000004                          RBU=IPL+2               ;CRB UNLOADER POINTER
000006                          LCW=RBU+2               ;LINE CONTROL WORD
000010                          LSW=LCW+2               ;LINE STATUS WORD
000012                          RSW=LSW+2               ;RECEIVE STATUS WORD
000014                          TSW=RSW+2               ;TRANSMIT STATUS WORD
000016                          ASX=TSW+2               ;ASSIGN SLOT INDEX
000020                          RCX=ASX+2               ;RECEIVE SLOT INDEX
000022                          ARA=RCX+2               ;ACTIVE RECEIVE ADDRESS
000024                          ABC=ARA+2               ;ACTIVE BYTE COUNT
000026                          UBL=ABC+2               ;ONE 2-WORD SLOT PER SUPPORTED BUFFER
000062                          IRC=UBL+MBX             ;INTERIM RECEIVE COUNT
000064                          TXC=IRC+2               ;USER CHARS XMITTED SO FAR
000066                          TSA=TXC+2               ;ORIGINAL USER XMIT ADDR
000070                          TBC=TSA+2               ;ORIGINAL USER XMIT COUNT
000072                          ISC=TBC+2               ;IDLE SYNCH CONTROL-MUST PRECEDE T1A
000074                          T1A=ISC+2               ;TRANSMIT BUFFER 1
000076                          T1C=T1A+2               ;TRANSMIT COUNT 1
000100                          T2A=T1C+2               ;TRANSMIT BUFFER 2
000102                          T2C=T2A+2               ;TRANSMIT COUNT 2
000104                          T3A=T2C+2               ;TRANSMIT BUFFER 3
000106                          T3C=T3A+2               ;TRANSMIT COUNT 3
000110                          TTM=T3C+2               ;TERMINATES TRANSMIT CHAIN (ALWAYS ZERO)
000112                          TBS=TTM+2               ;TAP BUFFER START
000114                          TRC=TBS+2               ;RECORD COUNTER (BYTE)
000115                          MRC=TRC+1               ;MASTER RECEIVE TIMER COUNTER (BYTE)
000116                          THC=TRC+2               ;HOLD COUNT
000120                          TCT=THC+2               ;TICK COUNT FOR CRB SERVICE TIMER REQUEST
000121                          ECH=TCT+1               ;CURRENT END CHARACTER-MUST PRECEDE CBC
000122                          CBC=ECH+1               ;COMPUTED BCC
000124                          RBC=CBC+2               ;RECEIVED BCC
000126                          BCC=RBC+2               ;BCC CHARACTER COUNT (INCLUDING PAD)
000127                          BCL=BCC+1               ;BCC LATCH
000130                          TCP=BCC+2               ;TRANSMIT CHAIN POINTER
000132                          CCT=TCP+2               ;ACTIVE CODE CONTROL TABLE
000134                          CAK=CCT+2               ;ACK RECOGNITION WORD (RCV)
000136                          AKT=CAK+2               ;XMIT ACK COUNTER (BYTE)
000137                          RIB=AKT+1               ;RECORDS-IN-BLOCK COUNTER (BYTE)
000140                          EKT=AKT+2               ;ENQ COUNTER (BYTE)
000141                          RKT=EKT+1               ;RETRY COUNTER (BYTE)
000142                          TCW=EKT+2               ;TERMINAL STATUS WORD
000144                          TT2A=TCW+2              ;TEMP. T2A
000146                          TT2C=TT2A+2             ;TEMP. T2C
000150                          TT3A=TT2C+2             ;TEMP. T3A
000152                          LGC=TT3A+2              ;COUNTER FOR LEADING GRAPHICS (REF. AS A BYTE)
000153                          SRC=LGC+1               ;SLAVE RECEIVE TIMER COUNTER (BYTE)
                                ;THE FOLLOWING IS THE TAP BUFFER STORAGE FOR THIS LINE
                                ;IT MUST BE THE LAST ITEM IN THE LINE TABLE
000154                          TIB=LGC+2

                                ; THE FOLLOWING DEFINES THE TABLE SIZE FOR LINIT SERVICE
000776                          LTL=TIB+BSIZE+2         ;--------------------------------------
000776                          . BLTL=LTL              ;MAKE THE SCIP HAPPY BY MAKING TABLE SIZE
                                                        ;A GLOBAL PARAMETER
```

445

```
                                ; LINE CONTROL WORD BIT DEFINITIONS
000001                          XRC=1           ;CURRENT CODE: 00 - ASCII,01 - SBT,10 - EBCDIC
000002                          X.=XRC+1        ;SECOND BIT OF CODE FLAG
000004                          MCF=4           ;MASTER CONTENTION FLAG
000010                          SIS=10          ;SERIAL/INTERLEAVED SELECT
000020                          MSS=20          ;MASTER/SLAVE SELECT
000100                          OOF=100         ;OPTIONS ONLY FLAG (PUTTC)
000200                          DLEF=200        ;DLE FLAG - PREVIOUS CHAR WAS DLE
000400                          INI=400         ;INIT FLAG - 0 IF INIT, 1 IF MESSAGE TRANSFER
001000                          NAT=1000        ;NATIVE MODE SELECTED
002000                          AFX=2000        ;ABNORMAL FORMAT XMIT
004000                          ITS=4000        ;ITB SELECT
010000                          XXP=10000       ;XMIT TRANSPARENT - TOGGLED FROM XPT
020000                          XPT=20000       ;TRANSPARENCY TOGGLE - 1ST PUTD OF BLOCK SETS XXP
100000                          RCVT=100000     ;RECEIVE TRANSPARENT - SET BY RECEIVED DLE STX

                                ; LINE STATUS WORD BIT DEFINITIONS
000001                          RBA=1           ;RECEIVE BUFFER ASSIGNED
000002                          CSA=2           ;CRB SERVICE ACTIVE
000004                          AAF=4           ;ACK AVAILABLE FLAG - UAP HAS DONE AN ACK ASRBUF
                                                ;SINCE THE LAST RECEIVED RECORD
000020                          STXF=20         ;STX FOUND IN TEXT - EDIT 17A
000040                          RJT=40          ;PUTD OVERFLOW REJECT FLAG
000200                          BCCA=200        ;BCC ACCUMULATE FLAG
000400                          SRTA=400        ;SLAVE RCVE TIMER ACTIVE
001000                          RDBF=1000       ;RESTORE X-MIT POINTER FLAG
002000                          ITBA=2000       ;ITB ACTIVE - PREVIOUS BLOCK ENDED WITH ITB
004000                          NEF=4000        ;NAK EXPECTED FLAG (TTD SEQUENCE SENT) - EDIT 17
010000                          XAC=10000       ;TRANSMIT ACTIVE
020000                          AEF=20000       ;ACK EXPECTED FLAG
040000                          ANF=40000       ;ACKNOWLEDGE NOTIFY FLAG
100000                          MIP=100000      ;MESSAGE IN PROGRESS
```

```
                                ; RECEIVE STATUS WORD BIT DEFINITIONS

000001                          CRD=1           ;CARRIER DROP FLAG
000002                          NUB=2           ;NO USER BUFFER ASSIGNED
000004                          DCH=4           ;DATA CHECK (BCC ERROR)
000010                          PCH=10          ;PARITY CHECK
000020                          UOVR=20         ;USER BUFFER OVERFLOW
000040                          COVR=40         ;CRB OVERFLOW
000100                          ROVR=100        ;RECEIVE OVERFLOW (HARDWARE)
000200                          SRTOF=200       ;SLAVE RCVE TIMEOUT FLAG

000400                          FBF=400         ;'FLUSH BUFFER' FLAG
001000                          TRNREC=1000     ;TRANSPARENT RECEORD RECEIVED (DLE-STX)
002000                          C.BEOF=2000     ;END OF TRANSMISSION (EOT DETECTED)
004000                          DSC=4000        ;DISCONNECT FLAG (DLE EOT RECEIVED)
010000                          MEND=10000      ;MESSAGE END (ETX RECEIVED)
020000                          BEND=20000      ;BLOCK END (ETB RECEIVED)
040000                          AEND=40000      ;ABNORMAL END - NO RECOGNIZABLE END CHAR
100000                          ARF=100000      ;ABNORMAL RECEIVE FORMAT ( NOT STX OR DLE STX)


                                ;TRANSMIT STATUS WORD BIT DEFINITIONS

000100                          MTOUT=100               ;MASTER TIME OUT HAS OCCURRED
000200                          OVXM=200                ;OVERRIDE TRANSMIT ATTEMPT


000400                          ETXF=400                ;ETX SENT
001000                          ESNT=1000               ;EOT SENT
002000                          BSNT=2000               ;BID (ENQ) MESSAGE SENT


040000                          RVI=40000               ;REVERSE INTERRUPT RECEIVED
100000                          NAKF=100000             ;NAK FLAG (ALSO SET BY ABNORMAL CONDITIONS)
```

```
                                ; CODE CONTROL TABLE DEFINITIONS

000000                          CRVI=0                  ;RVI
000001                          CWAK=CRVI+1             ;WACK
000002                          CPAD=CWAK+1             ;PAD
000003                          CSTX=CPAD+1             ;STX
000004                          CSOH=CSTX+1             ;SOH
000005                          CETB=CSOH+1             ;ETB ET CETERA
000006                          CETX=CETB+1
000007                          CITB=CETX+1
000010                          CDLE=CITB+1             ;DLE - ALSO START OF ACK MESSAGES
000016                          CSYN=CDLE+6             ;SYN - 4 OF THEM HERE FOR HEADER MESSAGE
000022                          CENQ=CSYN+4
000024                          CNAK=CENQ+2
000026                          CEOT=CNAK+2
000027                          BTCE=CEOT+1             ;NO OF CHARS IN BCC (ETB OR ETX OPERATION)
000030                          BTCI=BTCE+1             ;NO OF CHARS IN BCC (ITB OPERATION)
                                        .IF DF  R3780
                                CIRS=BTCI+1
                                        .ENDC

000031                          LASBYT=BTCI+1           ;USED TO FORCE NEXT INDEX EVEN

000032                          SYP=LASBYT&1+LASBYT     ;SYN POINTER (MUST BE WORD BOUNDARY)
000034                          KGC=SYP+2               ;KG11 CLEAR/SELECT CODE
000036                          ATT=KGC+2               ;ACTIVE TRANSMIT TRANSLATOR
000040                          ATR=ATT+2               ;ACTIVE RECEIVE TRANSLATOR
000042                          AMP=ATR+2               ;ACK MESSAGE POINTER
000044                          NMP=AMP+2               ;NAK MESSAGE POINTER
000046                          PPD=NMP+2               ;WHERE TO PUT THE PAD FOR BCC
000050                          MDC=PPD+2               ;MODE WORD FOR ISR MODE CONTROL (PUTTC - INIT)
000052                          CAK0=MDC+2              ;CHAR FOR ACK0 RECOGNITION
000053                          CAK1=CAK0+1             ;;CHAR FOR ACK1 MESSAGE
```

```
                                        .IF DF  RSX
000000                                  .PSECT  ISR,CON
                                        .ENDC
                                        .IF NDF OS
                                        .CSECT
                                        .ENDC

000000                                  ASCCCT=0


                                ;THE FOLLOWING IS USED FOR CHECKING CRB ENTRY
                                ;FOR CORRECT PARITY IF ASCII IS SELECTED

010000                          PARITY=10000
```

```
                                        .IF NZ CA.EBC

                                ;EBCDIC CODE CONTROL TABLE
000000      174                 EBCCCT: .BYTE 174               ;X'7C' = RVI FOLLOWING DLE
000001      153                         .BYTE 153               ;',' = WACK FOLLOWING DLE
000002      377                         .BYTE 377               ;PAD
000003      002                         .BYTE 02                ;STX
000004      001                         .BYTE 01                ;SOH
000005      046                         .BYTE 46                ;ETB
000006      003                         .BYTE 03                ;ETX
000007      037                         .BYTE 37                ;ITB
000010      020     160     377 EBCACK: .BYTE 20,160,377        ;DLE - ALSO ACK0 MESSAGE
000013      020     141     377         .BYTE 20,141,377        ;ACK1 MESSAGE
000016      062     062     062 EBCSYN: .BYTE 62,62,62,62       ;SYN SYN SYN SYN
000021      062
000022      055     377                 .BYTE 55,377            ;ENQ PAD
000024      075     377         EBCNAK: .BYTE 75,377            ;NAK PAD
000026      067                         .BYTE 67                ;EOT
000027      004                         .BYTE 04                ;COUNT FOR END CHAR + BCC + PAD
000030      003                         .BYTE 03                ;COUNT FOR BCC + PAD ONLY
                                        .IF DF  R3780
                                        .BYTE   36              ;IRS
                                        .ENDC

                                        .EVEN
000032  000016'                        .WORD EBCSYN            ;POINTER TO MESSAGE HEADER STRING
000034  000101                         .WORD 101               ;KG11 SELECT CODE (CRC16)
000036  001752'                        .WORD EBCXTR            ;EBCDIC  XMIT TRANSLATOR
000040  004006'                        .WORD EBCRTR            ;EBCDIC RECEIVE TRANSLATOR
000042  000010'                        .WORD EBCACK            ;EBCDIC ACK MESSAGE
000044  000024'                        .WORD EBCNAK            ;EBCDIC NAK MESSAGE
000046  000124                         .WORD RBC               ;LOCATION FOR PAD IN BCC STRING
000050  177051                         .WORD 177051            ;FOR ISR MODE CONTROL (PUTTC - INIT)
000052      160                        .BYTE 160               ;HEX 70 FOR ACK0 MESSAGE
000053      141                        .BYTE 141               ;HEX 61 FOR ACK1 MESSAGE
                                ;END OF EBCDIC CODE CONTROL TABLE

                                        .ENDC
                                        .IF Z   CA.EBC
                                        EBCCCT=0
                                        .ENDC
```

```
                                        SBTCCT=0
                                        .IF DF  RSX
                                        .PSECT  XXX
                                        .ENDC
```

```
                                ; CODE FOR USER INTERFACE GETS QUEUE ENTRIES.

000003                                  C.BGBC=3                        ; BASE CODE FOR
                                                                        ;GETS QUEUE ENTRIES
000303                          C.BGRD= C.BGBC+300      ;RECEIVE DONE CODE (4 WORD UIB ENTRY)
000304                          C.BGTD= C.BGBC+301      ;TRANSMIT DONE CODE (4 WORD UIB ENTRY)
000105                          C.BGMS= C.BGBC+102      ;MODEM STATUS CODE (2 WORD UIB ENTRY)


                                ; FUNCTION CODE ASSIGNMENTS FOR DEVICE
                                ,TYPE MATRIX SWITCH CALLS.

000000                                  C.BTSC=0                        ;TRANSMIT STARTUP FUNCTION CODE
000001                                  C.BMMC=1                        ;MODEM STATUS
000002                                  C.BMDC=2                        ;ISR MODE CONTROL
```

```
                                        ;RCVE DONE GETS PARAMETER LIST.
                                                .IF DF RSTS
                                                .GLOBL  C.BRMC
                                                .IFF
                                        ;
000010  000000                          C.BRMC: 0
000012  000000                                  0
000014  000000                                  0
000016  000000                                  0
                                                .ENDC


                                        .IF NDF RSX11M
                                        .EOT
                                        ;;;;;;;;;;;;;;;;;;;;;;;;;;
                                        ;END OF TAPE 1
                                        .ENDC
```

```
                                ;BISYNC TAPE 2 OF 5
                                ;LINIT COMMAND SERVICE ROUTINE

                                ;ENTERED FROM EMT COMMAND HANDLER VIA TERMINAL TYPE MATRIX
                                ;SWITCH WITH THE FOLLOWING REGISTERS PRESET:
                                ;R1 - POINTER TO LINIT PARAMETER LIST
                                ;R2 - POINTER TO START OF ASSIGNED LINE TABLE SPACE
                                ;R3 - LINE # AS A WORD INDEX
                                ;WORD 1 OF THE LINE TABLE HAS ALREADY BEEN LOADED WITH
                                ;THE DEVICE AND TERMINAL TYPE CODES.

000020                          C.BINT: ;MOV    R2,     R4
                                        ;ADD    #LTL,   R4              ;POINTER TO NEXT AVAILABLE
                                                                        ;WORD IN RESERVED TABLE AREA

                                ;INITIALIZE LINE TABLE
                                        ;MOV    R4,     4(R1)           ;STORE POINTER TO NEXT
                                                                        ;AVAILABLE WORD IN
                                                                        ;LINE TABLE AREA.
                                        ;MOV    R2,R5                   ;NOW CLEAR THE TABLE
                                        ;ADD    #2,R5
                                ;INTCLR:        CLR     (R5)+
                                        ;CMP    R5,R4
                                        ;BNE    INTCLR
000020  112762  000007  000152          MOVB    #LDGRPH,LGC(R2) ;INITIALIZE LEADING GRAPHICS COUNTER
                                        .IF DF  UAPTIM
000026  010104                          MOV     R1,R4                   ;POINTER TO PARAMETER LIST
000030  062704  000006                  ADD     #6,R4                   ;POINT TO TIMER VALUES
000034  012167  177740                  MOV     (R4)+,CONTIM            ;SAVE CONTENTION TIMER
000040  012467  177742                  MOV     (R4)+,TICINT            ;SAVE TIMER INTERVAL
000044  012467  177732                  MOV     (R4)+,MSRT              ;SAVE MASTER TIMER
000050  012467  177730                  MOV     (R4)+,SSRT              ;SA'E SLAVE TIMER
000054  011462  000120                  MOV     (R4),TCT(R2)            ;S_ _ CRB TIMER
                                        .IFF
                                        MOVB    6(R1),TCT(R2)           ;SET THE TICK COUNT FOR THIS LINE
                                        .ENDC
000060  105062  000115                  CLRB    MRC(R2)         ;RESET MASTER RECEIVE TIMER
000064  004767  004332                  JSR     PC,CSSRT        ;RESET SLAVE RECEIVE TIMER
000070  004767  004514                  JSR     PC,INTVTM       ;START 1/2 SECOND INTERVAL TIMER
000074  010204                          MOV     R2,R4
000076  062704  000072                  ADD     #ISC,R4                 ;SET POINTER TO TRANSMIT CHAIN
000102  010462  000130                  MOV     R4,TCP(R2)
000106  010204                          MOV     R2,R4                   ;COMPUTE TAP BUFFER ADDRESS...
000110  062704  000154                  ADD     #TIB,R4                 ;... FOR THIS LINE
000114  010462  000112                  MOV     R4,TBS(R2)              ;AND STORE IN LINE TABLE
000120  000250                          CLN
000122  000207                          RTS     PC                      ;N CONDITION CODE BIT CLEAR.

                                ;RECEIVE DONE PROCESSOR
                                ;RELEASE BUFFER TO USER VIA. GETS QUEUE


000124  016205  000020          C.BRCD: MOV     RCX(R2),R5              ;LOCATE CURRENT RCV SLOT
000130  06?205                          ADD     R2,R5                   ;NOW HAVE BASE ADDRESS OF BUFFER LIST
000132  062705  000026                  ADD     #UBL,R5
000136  011567  177650                  MOV     (R5),C.BRMC+2           ;MOVE ADDRESS TO UIB ENTRY
000142  005025                          CLR     (R5)+                   ;AND MARK SLOT FREE
```

```
 703 000144  011567  177644                    MOV     (R5),C.BRMC+4           ;SAME FOR ORIG. BYTE COUNT
 704 000150  166267  000024  177636            SUB     ABC(R2),C.BRMC+4        ;BUT ADJUST TO ACTUAL COUNT
 705 000156  005025                            CLR     (R5)+
 706                                ; NOW UPDATE THE RECEIVE SLOT INDEX
 707 000160  062762  000004  000020            ADD     #4,RCX(R2)              ;ENTRIES ARE TWO WORDS
 708 000166  026227  000020  000034            CMP     RCX(R2),#MBX            ;TIME TO RESET INDEX?
 709 000174  002404                            BLT     C.BRCZ          ;NOT YET
 710 000176  005062  000020                    CLR     RCX(R2)                 ;YES
 711 000202  162705  000034                    SUB     #MBX,R5                 ;RESET R5 TO BASEOF LIST
 712                                ; NOW ACTIVATE NEXT BUFFER, IF ANY
 713 000206  005715                C.BRCZ: TST     (R5)                    ;IS ANOTHER BUFFER AVAILABLE?
 714 000210  001003                            BNE     C.BRCY
 715 000212  042762  000001  000010            BIC     #RBA,LSW(R2)            ;NONE AVAILABLE (SLOT EMPTY)
 716 000220  012562  000022        C.BRCY: MOV     (R5)+,ARA(R2)           ;MOVE ADDRESS (OR ZERO)
 717                                            .IF DF RSTS
 718                                            JSR     PC,GTBF1                ;GET MAPPED ADDRESS FOR BUFFER
 719                                            .ENDC
 720 000224  011562  000024                    MOV     (R5),ABC(R2)            ;SAME FOR COUNT
 721                                ; NOW GO RECONDITION THE DP11 ISR
 722 000230  032762  002000  000010            BIT     #ITBA,LSW(R2)           ;IF ITB ACTIVE...
 723 000236  001011                            BNE     C.BCR3                  ;....LEAVE DP11 ALONE
 724 000240  012767  170000  001540            MOV     #170000,C.BIMS+2        ;REESTABLISH STRIP SYNC
 725 000246  012700  000002                    MOV     #C.BMDC,R0              ;;
 726 000252  012701  002004'                   MOV     #C.BIMS,R1
 727 000256  004767  000000G                   JSR     PC,C.SARP
 728 000262  016267  000012  177526  C.BCR3: MOV     RSW(R2),C.BRMC+6        ;STATUS BITS
 729 000270  105762  000012                    TSTB    RSW(R2)                 ;IF ANY ERRORS....
 730 000274  001406                            BEQ     C.BNXT
 731 000276  152767  000200  177512            BISB    #200,C.BRMC+6                   ;..SET COMBINED ERROR FLAG
 732 000304  042762  000020  000012            BIC     #UOVR,RSW(R2)   ;**********MAKE SURE OVRFL SWTCH ALWAYS RESET
 733 000312  105067  177472        C.BNXT: CLRB    C.BRMC          ;LINE NUMBER ALWAYS 0
 734 000316  112767  000303  177465            MOVB    #C.BGRD,        C.BRMC+1        ;RCVE COMPLETE TYPE CODE
 735 000324  012705  000010'                   MOV     #C.BRMC,R5
 736 000330  004767  000000G                   JSR     PC,C.SFY                ;STORE RCVE MESSAGE
 737 000334  032762  000400  000006            BIT     #INI,LCW(R2)    ;MESSAGE X-FER?
 738 000342  001403                            BEQ     C.BCR1          ;NO
 739
 740                                                                    ;COMPLETE ENTRY
 741 000344  042762  000004  000010            BIC     #AAF,LSW(R2)            ;RECEIVED A NEW RECORD SO OLD
 742                                                                    ;ACK IS NO LONGER AVAILABLE
 743 000352  032762  002000  000010  C.BCR1: BIT     #ITBA,LSW(R2)           ;IF ITB, DON'T CLEAR STATUS
 744 000360  001002                            BNE     C.BCR2
 745 000362  005062  000012                    CLR     RSW(R2)         ;
 746 000366  000207                C.BCR2: RTS PC
```

```
                                        ;MODEM STATUS UIB ENTRY PROCESSOR
                                        ;PROCESSING ROUTINE. ENTERED FROM BUFFER UNLOADER WITH:
                                        ;R0 CONTAINING LINE# AND TYPE CODE (C.BMSC=BASE TYPE CODE+1)
                                        ;R1 CONTAINING MODEM STATUS WORD.
                                        ;R2 CONTAINING POINTER TO BASE OF LINE TABLE.
                                        ;R3 CONTAINING LINE# AS WORD INDEX.
000370                                  C.BMSE:
000370  000207                                RTS     PC
                                        ; PUTD EMT COMMAND SERVICE ROUTINE.
                                        ;ENTER FROM TERMINAL TYPE MATRIX SWITCH WITH:
                                        ;R0:    FUNCTION CODE = 000 - NORMAL BLOCK
                                        ;                       040 - LAST BLOCK
                                        ;                       100 - ABNORMAL FORMAT
                                        ;R1:    POINTER TO PUTD PARAMETER LIST
                                        ;R2:    POINTER TO FIRST WORD OF LINE TABLE
                                        ;R3:    LINE #*2

000372  105721                         C.BPD:  TSTB    (R1)+           ;SKIP THE FIRST BYTE IN LIST
000374  032762  010000  000010                BIT     #XAC,LSW(R2)            ;IS LINE ALREADY BUSY
                                                                        ;TRANSMITTING?
000402  001406                                BEQ     PUTD11
                                        ;LINE ALREADY BUSY.  RETURN WITH 'N' CONDITION CODE BIT SET.
000404  042762  000040  000010          C.BPDR: BIC     #RJT,LSW(R2)    ;CLEAR THE REJECT FLAG
000412  000242                                CLV
000414  000270                          C.BPDX: SEN

000416  000477                                BR      PTNEXT
000420  112100                          PUTD11: MOVB    (R1)+,R0            ;GET PUTD OP CODE.
000422  005761  000002                  PUTD1A: TST     2(R1)               ;POSITIVE NON-ZERO COUNT?
000426  003772                                BLE     C.BPDX  ;NO - REJECT PUTD
000430  052762  060000  000010                BIS     #AEF!ANF,LSW(R2)
000436  042762  002000  000006                BIC     #AFX,LCW(R2)
000444  112762  000010  000141                MOVB    #NAKCNT,RKT(R2)         ;SET RETRY COUNT
000452  005062  000064                        CLR     TXC(R2)                 ;CLEAR TOTAL XMIT COUNT
000456  032762  002000  000010                BIT     #ITBA,LSW(R2)
000464  001012                                BNE     PUTD1
000466  032762  020000  000006                BIT     #XPT,LCW(R2)
000474  001406                                BEQ     PUTD1
000476  052762  010000  000006                BIS     #XXP,LCW(R2)
000504  042762  020000  000006                BIC     #XPT,LCW(R2)
000512  032700  000100          PUTD1:  BIT #100,R0             ;IS IT ABNORMAL FORMAT (100)
000516  001406                          BEQ PTNORM              ;NO - PROCEED
000520  052762  002000  000006          BIS #AFX,LCW(R2)                        ;YES - SET FLAG
000526  042762  022000  000010          BIC     #AEF!ITBA,LSW(R2)       ;NOT EXPECTING ACK - UNLESS ENQ
                                                                        ;  IS SENT
000534  012162  000066          PTNORM: MOV     (R1)+,TSA(R2)           ;NO.  SAVE USER ADDRESS AND COUNT
000540  011162  000070                  MOV     (R1),TBC(R2)
000544  016201  [illegible]             MOV CCT(R2),R1          ;SET POINTER TO CHAR CONTROL TABLE
000550  116162  [illegible]  000121      MOVB    CETB(R1),ECH(R2)        ;PROVISIONALLY SET END CHAR TO ETB
000556  032700  [illegible]              BIT #40,R0              ;LAST BLOCK REQUEST (40)
000562  001406                          BEQ PTN0
000564  116162  000006  000121          MOVB CETX(R1),ECH(R2)           ;YES - CHANGE END CHAR TO ETX
000572  052762  000400  000014          BIS #ETXF,TSW(R2)               ;SET 'ETX SENT' FLAG
000600  004767  000504          PTN0:   JSR PC,PUTDO                    ;GO PROCESS USER BUFFER
000604  032762  000040  000010          BIT #RJT,LSW(R2)                ;TAKE REJECT EXIT?
000612  001002                          BNE PTN1                        ;YES
000614  000250                          CLN
```

```
805 000616  000207                         PTNEXT: RTS     PC                  ;RETURN TO EMT HANDLER
806 000620  042762  000400  000014  PTN1:   BIC     #ETXF,TSW(R2)       ;RESET 'ETX SENT' FLAG
807 000626  000666                          BR      C.BPDR
808
```

```
                                ;PUTD CHARACTER BUFFERING ROUTINE
000630                          PTR1:   .IF DF LSI11
                                        MOVB    R0,CRCCHR               ;ACCUMULATE IN BCC
                                        JSR     PC,$CRC16               ;USING SOFTWARE ROUTINE
                                        .IFF
000630  110037  170704                  MOVB R0,@#KGDATA                ;ACCUMULATE IN BCC
                                        .ENDC
000634  110024                  PTR3:   MOVB R0,(R4)+                   ;STORE IN TAP BUFFER
000636  005262  000102                  INC T2C(R2)                     ;INCREMENT CURRENT COUNT
000642  005262  000064                  INC TXC(R2)                     ; INCR. TOTAL COUNT
000646  026227  000102  000615          CMP T2C(R2),#BSIZE-3            ;TAP BUFFER LIMIT REACHED?
000654  002001                          BGE PTR2                        ;YES
000656  000207                          RTS PC

                                ; THE TAP INTERNAL BUFFER HAS OVERFLOWED -
                                ; DISPATCH PREVIOUS RECORDS AND REJECT THE CURRENT ONE
000660  005726                  PTR2:   TST (SP)+                       ;DISCARD THE JSR RETURN ADDRESS
000662  105762  000114                  TSTB TRC(R2)                    ;IS THIS FIRST BUFFER?
000666  003003                          BGT PTR4                        ;NO - OK TO PROCEED
000670  005726                          TST     (SP)+                   ;DISCARD JSR PUTDO
                                                                        ;RETURN ADDRESS
000672  000167  177506                  JMP C.BPDR                      ;TAKE REJECT EXIT
000676  016262  000116  000102  PTR4:   MOV THC(R2),T2C(R2)             ;RESET THE TRANSMIT PARAMETERS
                                ;NOW WE HAVE TO RECOMPUTE THE BCC FOR THE PREVIOUS RECORD
                                ; USING ETB INSTEAD OF ITB AS THE END CHAR
                                        .IF DF LSI11
                                        MOV     CBC(R2),CRCACC          ;RESTORE THE PRE-ITB VALUE OF BCC
                                        .IFF
000704  012737  000133  170700          MOV #133,@#KGSTAT               ;RESET KG11 AND SELECT LRC16
000712  016237  000122  170704          MOV CBC(R2),@#KGDATA    ;RESTORE THE PRE-ITB VALUE OF BCC
                                        .ENDC
000720  116162  000005  000121          MOVB    CETB(R1),ECH(R2)        ;USE ETB AS END CHAR
000726  052762  000040  000010          BIS #RJT,LSW(R2)                ;SET FOR REJECT EXIT
                                        .IF NDF LSI11
000734  016137  000034  170700          MOV KGC(R1),@#KGSTAT            ;NOW SELECT CURRENT BCC MODE
                                        .ENDC
000742  000167  000634                  JMP PUTD6               ;NOW GO OUTPUT THE BUFFER
```

```
                                        ; PUTD PROCESSOR (CONTINUED)

                                        ;PROCESS A SPECIAL CHAR
000746  116303  003652'        PTBSP:  MOVB ACTTAB(R3),R3              ;FETCH THE ACTION CODE
000752  000173  003762'                 JMP @PTSAT(R3)                  ;GO TO ACTION ROUTINE

                                        ;PROCESS A DLE
000756  032762  010000  000006 PTDLE:  BIT #XXP,LCW(R2)                 ;TRANSPARENCY?
000764  001410                         BEQ PTDL1                       ;NO.  TREAT AS TEXT
000766  032762  002000  000006         BIT #AFX,LCW(R2)                ;IF ABNORMAL FORMAT ...
000774  001004                         BNE PTDL1                       ;TAKE NO SPECIAL ACTION ON CHAR
000776  116100  000010                 MOVB CDLE(R1),R0                ;YES - STUFF A DLE AHEAD OF IT
001002  004767  177626                 JSR PC,PTR3
001006  000167  000526         PTDL1:  JMP PUTD5

                                        ;PROCESS AN ITB
001012                          PTITB:
001012  032762  010000  000006         BIT     #XXP,LCW(R2)            ;TRANSPARENCY?
001020  001407                         BEQ     PTIT3                   ;NO - ACCEPT ITB
001022  005362  000070                 DEC     TBC(R2)                 ;YES - IS THIS THE LAST CHAR?
001026  001404                         BEQ     PTIT3                   ;YES - ACCEPT ITB
001030  004767  177574                 JSR     PC,PTR1                 ;NO - TREAT AS NORMAL DATA
001034  000167  000456                 JMP     PUTD4                   ;PROCESS NEXT CHARACTER
001040  105262  000114         PTIT3:  INCB TRC(R2)                    ;COUNT NO OF RECORDS IN BLOCK
001044  032762  010000  000006         BIT #XXP,LCW(R2)                ;TRANSPARENCY?
001052  001404                         BEQ PTIT1
001054  116100  000010                 MOVB CDLE(R1),R0                ;YES - STUFF A DLE IN XMIT BUFFER
001060  004767  177550                 JSR PC,PTR3
001064  016262  000102  000116 PTIT1:  MOV T2C(R2),THC(R2)              ;THIS IS COUNT IF NEXT RECORD REJECTED
                                        .IF DF LSI11
                                        MOV     CRCACC,CBC(R2)          ;BCC FOR REJECT CASE
                                        .IFF
001072  013762  170702  000122         MOV @#KGBCCR,CBC(R2)            ;BCC FOR REJECT CASE
                                        .ENDC
001100  126227  000114  000007         CMPB TRC(R2),#C.BUFF            ;MAX NO. OF RECORDS?
001106  002004                         BGE     PTIT4A                  ;YES-GO OUTPUT IT
001110  026227  000102  000612         CMP     T2C(R2),#BSIZE-6                ;BCC GOING TO BE TOO NEAR END
001116  002402                         BLT PTIT4                       ;NO - PROCEED
001120  000167  000456         PTIT4A: JMP PUTD6                       ;YES - GO OUTPUT IT
001124  116100  000007         PTIT4:  MOVB CITB(R1),R0                ;RESTORE THE ITB CHAR
001130  004767  177474                 JSR PC,PTR1                     ;GO BUFFER CHARACTER
001134  052762  002000  000010         BIS #ITBA,LSW(R2)               ;SET ITB ACTIVE
                                        .IF DF LSI11
                                        MOVB    CRCACC,R0       ;MOVE BCC TO TAP BUFFER
                                        .IFF
001142  113700  170702                 MOVB @#KGBCCR,R0                ; MOVE BCC TO TAP BUFFER
                                        .ENDC
001146  004767  177462                 JSR PC,PTR3
                                        .IF DF LSI11
                                        MOVB    CRC[illegible]C+1,R0            ;YES - MOVE SECOND BYTE
                                        .IFF
001152  113700  170703                 MOVB @#KGB[illegible]1,R0               ;YES - MOVE SECOND BYTE
                                        .ENDC
001156  004767  177452                 JSR PC,PTR3
001162  000167  000456         PTIT2:  JMP PUTD8
                                        ;PROCESS AN STX IN THE USE MESSAGE
001166  052762  000020  000010 PTSTX:  BIS     #STXF,LSW(R2)           ;SET STX FOUND FLAG - EDIT 17
```

```
 907 001174  000561                          BR      PUTD5                   ;PROCESS NORMALLY
 908                                  ;
 909
 910                                  ; PROCESS AN ENQ IN THE USER MESSAGE
 911 001176  032762  002000  000006  PTENQ:  BIT #AFX,LCW(R2)                ;IF NOT ABNORMAL FORMAT..
 912 001204  001555                          BEQ PUTD5                       ;TREAT NORMALLY
 913 001206  032762  000020  000010          BIT     #STXF,LSW(R2)           ;HAS AN STX BEEN FOUND? - EDIT 17
 914 001214  001404                          BEQ     PTENQ1                  ;NO - TREAT AS A BID - EDIT 17
 915 001216  052762  064000  000010          BIS     #NEF!AEF!ANF,LSW(R2)    ;TTD SEQUENCE - EXPECT NAK - EDIT 17
 916 001224  000545                          BR      PUTD5                   ;CONTINUE PROCESSING - EDIT 17
 917 001226  052762  060000  000010  PTENQ1: BIS #AEF!ANF,LSW(R2)            ;OTHERWISE SET 'ACK EXPECTED'
 918 001234  052762  002000  000014          BIS #BSNT,TSW(R2)               ;SET 'BID SENT'FLAG
 919 001242  000536                          BR PUTD5
 920
 921                                  ;PROCESS AN EOT IN THE USER MESSAGE
 922 001244  032762  002000  000006  PTEOT:  BIT     #AFX,LCW(R2)            ;ABNORMAL FORMAT
 923 001252  001532                          BEQ     PUTD5                   ;NO - TREAT AS NORMAL DATA
 924 001254  052762  001000  000014          BIS #ESNT,TSW(R2)               ;SET 'EOT SENT' FLAG
 925 001262  052762  000400  000006          BIS #INI,LCW(R2)                ;ALLOW XMIT WITHOUT INIT PHASE
 926 001270  000523                          BR PUTD5
 927
 928                                  ;PROCESS SYN IN THE USER PROGRAM MESSAGE
 929 001272  032762  010000  000006  PTSYN:  BIT     #XXP,LCW(P2)            ;TRANSPARENCY ACTIVE?
 930 001300  001117                          BNE     PUTD5                   ;YES - TREAT AS NORMAL DATA
 931 001302  004767  177326                  JSR     PC,PTR3                 ;BUFFER CHAR BUT DON'T
 932 001306  000516                          BR      PUTD5A                  ;INCLUDE IN BCC
```

```
                                        PUTDO:
001310                                  
001310  042762  177377  000014          BIC #177377,TSW(R2)             ;LEAVE ETX FLAG ALONE, RESET OTHERS
                                        .IF DF RSTS!LSI11
                                        .IF DF RSTS
                                        MOV     #121,@#KGSTAT           ;SET KG TO CURRENT MODE
                                        .ENDC
                                        .IF DF LSI11
                                        CLR     CRCACC          ;CLEAR THE BCC REGISTER
                                        .ENDC
                                        .IFF
001316  012737  000020  170700          MOV #20,@#KGSTAT                ;CLEAR THE BCC REGISTER...
001324  016137  000034  170700          MOV KGC(R1),@#KGSTAT            ;...AND INIT THE KG11 FOR CURRENT CODE
                                        .ENDC
001332  032762  002000  000010          BIT #ITBA,LSW(R2)               ;ITB ACTIVE?
001340  001405                          BEQ PTDZ
001342  016204  000100                  MOV     T2A(R2),R4      ;INIT FOR NON-FIRST RECORD OF ITB BLOCK
001346  066204  000102                  ADD     T2C(R2),R4      ;PICK UP WHERE WE LEFT OFF
001352  000406                          BR      PUTD3           ;  IN THE TAP BUFFER
001354  005062  000102          PTDZ:   CLR T2C(R2)                     ;CLEAR CURRENT COUNT
001360  016204  000112                  MOV TBS(R2),R4                  ;TAP'S TRANSMIT BUFFER
001364  010462  000100                  MOV  R4,T2A(R2)                 ;SET POINTER FOR TAP BUFFER
001370  016205  000066          PUTD3:  MOV TSA(R2),R5                  ;POINTER TO USER BUFFER
001374  032762  002000  000006          BIT     #AFX,LCW(R2)            ;ABNORMAL FORMAT?
001402  001045                          BNE     PUTD4                   ;YES BYPASS DLE/STX CODE
                                        .IF DF RSTS
                                        JSR     PC,GTBFAD               ;GO GET ADD IN CASE MAPPED OUTSIDE
                                                                        ;OF THE MONITOR
                                                                        ;UNLESS IS ABNORMAL FORMAT
                                        .ENDC
001404  032762  002000  000010          BIT #ITBA,LSW(R2)               ;1ST RECORD OF BLOCK?
001412  001406                          BEQ PUTD23                      ;DON'T PUT SYNS IN BUFFER THIS TIME
001414  116100  000016                  MOVB CSYN(R1),R0                ;PUT 2 SYNS IN TAP BUFFER
001420  004767  177210                  JSR PC,PTR3
001424  004767  177204                  JSR PC,PTR3                     ;SECOND SYN
001430  032762  010000  000006  PUTD23: BIT #XXP,LCW(R2)                ;NO - PUT STX THRU KG11
001436  001005                          BNE     PUTD24                  ;YES
001440  032762  002000  000010          BIT     #ITBA,LSW(R2)
001446  001023                          BNE     PUTD4
001450  000404                          BR      PUTD22
001452  116100  000010          PUTD24: MOVB CDLE(R1),R0                ;PRECEDED BY DLE IF TRANSPARENT
001456  004767  177146                  JSR PC,PTR1                     ;DLE GOES INTO BUFFER
001462  116100  000003          PUTD22: MOVB CSTX(R1),R0
001466  004767  177136                  JSR PC,PTR1
001472  032762  002000  000010          BIT #ITBA,LSW(R2)               ;  IF ITB NOT ACTIVE...
001500  001006                          BNE PUTD4
                                        .IF DF RSTS!LSI11
                                        .IF DF RSTS
                                        MOV     #121,@#KGSTAT           ;SET KG TO CURRENT CODE INITED
                                        .ENDC
                                        .IF DF LSI11
                                        CLR     CRCACC          ;CLEAR THE BCC REGISTER
                                        .ENDC
                                        .IFF
001502  012737  000020  170700          MOV #20,@#KGSTAT                ;WIPE (DLE)STX OUT OF KG
001510  016137  000034  170700          MOV KGC(R1),@#KGSTAT
                                        .ENDC
001516  005000                  PUTD4:  CLR R0                          ;GET USER CHAR WITHOUT SIGN EXT.
```

```
 991 001520 152500                          BISB (R5)+,R0
 992 001522 004771 000036                   JSR PC,@ATT(R1)              ;GO TO ACTIVE TRANSLATOR
 993 001526 120327 000037                   CMPB R3,#37                  ;IS ASCII <40?
 994 001532 003002                          BGT PUTD5
 995 001534 000167 177206                   JMP PTBSP                    ;IF SO TEST FOR SPECIAL ACTION
 996 001540 004767 177064          PUTD5:   JSR PC,PTR1                  ;GO BUFFER THE CHAR
 997 001544 005362 000070          PUTD5A:  DEC TBC(R2)                  ;IS THAT ALL THE USER CHARS?
 998 001550 003362                          BGT PUTD4                    ;NO - DO THE NEXT ONE
 999 001552 032762 002000 000006            BIT #AFX,LCW(R2)             ;IF ABNORMAL  FORMAT, BRANCH
1000 001560 001063                          BNE PTDAB
1001 001562 032762 010000 000006            BIT #XXP,LCW(R2)             ;TRANSPARENCY?
1002 001570 001404                          BEQ PUTD6                    ;NO.
1003 001572 116100 000010                   MOVB CDLE(R1),R0             ;YES - STUFF IN A DLE
1004 001576 004767 177032                   JSR PC,PTR3                  ;DON'T INCLUDE IN BCC
1005 001602                       PUTD6:    .IF DF LSI11
1006                                        MOVB    ECH(R2),CRCCHR       ;ACCUM END CHAR IN BCC
1007                                        JSR     PC,$CRC16            ;
1008                                        .IFF
1009 001602 116237 000121 170704            MOVB ECH(R2),@#KGDATA        ;ACCUM. END CHAR IN BCC
1010                                        .ENDC
1011 001610 010262 000104                   MOV R2,T3A(R2)               ;HAVEN'T SEEN ITB THIS PASS
1012 001614 062762 000121 000104            ADD #ECH,T3A(R2)                     ;SO SET BUF3 FOR ETB OR ETX
1013 001622 116162 000027 000106            MOVB BTCE(R1),T3C(R2)        ;SET BUF3 COUNT FOR BCC PAD
1014 001630 042762 002000 000010            BIC #ITBA,LSW(R2)            ;FOR LAST RECORD OF BLOCK
1015 001636                       PUTD7:    .IF DF LSI11
1016                                        MOV     CRCACC,CBC(R2)       ;SET BCC INTO XMIT STRING
1017                                        .IFF
1018 001636 013762 170702 000122            MOV @#KGBCCR,CBC(R2)         ;SET BCC INTO XMIT STRING
1019                                        .ENDC
1020 001644 005003                PUTD8:    CLR     R3                ;SET UP LINE NUMBER
1021 001646 032762 002000 000014            BIT #BSNT,TSW(R2)                ;IF MASTER INIT ....
1022 001654 001015                          BNE PUTD9                        ;..DON'T SEND IT YET
1023 001656 032762 002000 000010            BIT #ITBA,LSW(R2)                ;IF ITB ACTIVE, DON'T XMIT..
1024 001664 001015                          BNE PUTD10                       ;UNTIL ETB OR ETX TIME
1025 001666 010200                          MOV R2,R0                        ; POINTER TO PAD LOCATION
1026 001670 066100 000046                   ADD PPD(R1),R0
1027 001674 116110 000002                   MOVB CPAD(R1),@R0                ;AND PUT IN A PAD CHAR
1028 001700 004767 000150'                  JSR     PC,ACKOUT                ;INITIATE X-MIT
1029 001704 105062 000114                   CLRB TRC(R2)                     ;RESET RECORD COUNTER
1030 001710 052762 010000 000010  PUTD9:    BIS #XAC,LSW(R2)                 ;SET XMIT ACTIVE
1031                                        .IF DF RSTS
1032                              PUTD10:
1033                                        .ENDC
1034 001716 000207                          RTS PC
1035
1036                                        .IF NDF RSTS
1037 001720 005000                PUTD10:   CLR R0                   ;CREATE ACTUAL LINE NO.
1038 001722 004767 000532'                  JSR PC,TDENT                     ;GENERATE XMIT DONE UIB ENTRY
1039 001726 000207                          RTS PC                           ;AND DON'T SET XMIT ACTIVE YET
1040                                        .ENDC
1041
1042
1043                              ;FINAL SETUP FOR ABNORMAL FORMAT TRANSMIT
1044
1045 001730                       PTDAB:
1046                                        .IF DF  ASCI              ;IF ASCII INPUT INSERT A PAD
1047 001730 010162 000104                   MOV     R1,T3A(R2)               ;PUT IN PAD CHARACTER
```

```
1048 001734  062762  000002  000104          ADD     #CPAD,T3A(R2)
1049 001742  012762  000001  000106          MOV     #1,T3C(R2)              ;AND CHARACTER COUNT IN BUFFER
1050                                          .ENDC
1051                                          .IF NDF ASCI            ;IF NOT ASCII USER PROGRAM SUPPLIES PAD
1052                                          CLR     T3A(R2)                 ;TERMINATE CHARACTER STRING
1053                                          .ENDC
1054 001750  000735                          BR PUTD8
1055
```

```
1057
1058                                    ;HOLLERITH-EBCDIC TRANSMIT TRANSLATOR
1059                                            .IF NZ  CA.EBC
1060 001752                             EBCXTR.
1061
1062 001752  032762  001000  000006            BIT     #NAT,LCW(R2)            ;NATIVE MODE?
1063 001760  001403                             BEQ     EBCXT1                  ;NO - PERFORM NORMAL TRANSLATION
1064                                            ;TRANSLATE THROUGH EBCD TO ASCII TABLE
1065                                            ;FOR SPECIAL CODE TEST. DON'T TRANSLATE DATA CHARACTER
1066
1067 001762  116003  003052'                    MOVB    ETA(R0),R3
1068 001766  000403                             BR      X1                      ;BACK TO PUTD PROCESSOR
1069
1070                                            .IF DF  ASCI
1071 001770  110003                     EBCXT1: MOVB    R0,R3
1072 001772  116000  003452'                    MOVB    AEXB(R0),R0
1073 001776  042703  177600             X1:     BIC     #177600,R3              ;STRIP TO 7 BIT FOR ID
1074 002002  000207                             RTS     PC                      ;
```

```
1076                                    ;-PUTTC (PUT TERMINAL CONTROL)
1077                                    ;SERVICE ROUTINE.
1078                                    ;ENTER FROM TERMINAL TYPE MATRIX SWITCH WITH:
1079                                    ;R0:    FUNCTION CODE 2
1080                                    ;R1:    POINTER TO PUTTC PARAMETER LIST
1081                                    ;R2:    POINTER TO FIRST WORD OF LINE TABLE
1082                                    ;R3:    LINE #*2
1083
1084                                    ;PUTTC PARAMETER LIST FORMAT:
1085                                    ;LOCP:  BITS 0-7=LINE#, BITS 8-15=OPCODE 002
1086                                    ,LOCP+2:        MODE CONTROL WORD
1087                                    ;LOCP+4:        RESERVED FOR MODE STATUS WORD
1088
1089                                    ;PARAMETER LIST FORMAT FOR ISR MODE CONTROL AND SENSE CALL.
1090                                    ;R1 MUST POINT TO C.BIMS
1091                                            .IF DF RSTS
1092                                            .GLOBL  C.BIMS
1093                                            .IFF
1094                                    ;
1095                                    ;
1096                                    ;
1097 002004  000000                    C.BIMS: .WORD   0               ;ISR MODE CONTROL PARAMETER LIST
1098 002006  000000                            .WORD   0               ;ISR MODE CONTROL WORD
1099 002010  000000                            .WORD   0               ;RESERVED FOR ISR MODE STATUS.
1100 002012  000000                            .WORD   0               ;SYN CHAR FOR DP11 SYNC REG.
1101                                            .ENDC
1102
1103
1104                                    ;ENTER HERE FROM THE TERMINAL MATRIX SWITCH
1105
1106 002014  010146                    C.BPTC: MOV     R1,-(SP)        ;SAVE PARAM LIST POINTER FOR LATER
1107 002016  016104  000002                    MOV 2(R1),R4            ;FETCH THE MODE WORD
1108 002022  042762  000100  000006            BIC #00F,LCW(R2)        ;CLEAR THE OPTION-ONLY FLAG
1109 002030  016205  000006                    MOV LCW(R2),R5          ; FETCH CURRENT LCW
1110 002034  032704  040000                    BIT #40000,R4           ;WHAT TYPE OF PUTTC?
1111 002040  001002                            BNE PTC1
1112 002042  000167  000410                    JMP PTMORS              ;MESSAGE OR STATUS
1113                                    ; MUST BE INITIALIZATION OR TERMINATION
1114 002046  005704                    PTC1:   TST R4                  ;TERMINATION?
1115 002050  100002                            BPL PTC2
1116 002052  000167  000542                    JMP PTTERM              ;TERMINATION!
1117                                    ;THE FOLLOWING IS PROCESSOR FOR INITIALIZATION PUTTC
1118 002056  042762  177200  000142    PTC2:   BIC #177200,TCW(R2)     ;PREPARE THE TERMINAL CONTROL WORD
1119 002064  042705  176670                    BIC #176670,R5          ;PREPARE THE CONTROL AND STATUS WORDS
1120 002070  042762  102700  000010            BIC #102700,LSW(R2)
1121 002076  042704  000577                    BIC #577,R4             ;CLEAR THE UNWANTED BITS
1122 002102  050462  000142            PTC4:   BIS R4,TCW(R2)          ;NOW SAVE THE NEW SETTINGS
1123 002106  052705  000002                    BIS     #XRC+1,R5       ;SET CODE FOR EBCDIC
1124                                    ; SET LINE TABLE POINTER TO APPROPRIATE CODE CONTROL TABLE
1125 002112  010562  000006            TC13:   MOV R5,LCW(R2)          ;FIRST SAVE THE NEW LCW
1126 002116  012705  000000'                   MOV     #EBCCCT,R5      ;MOVE IN CODE INDICATOR
1127 002122  016200  000132                    MOV CCT(R2),R0          ;SAVE THE OLD CCT POINTER (EDIT 25)
1128 002126  016201  000134                    MOV CAK(R2),R1          ;AND THE OLD ACK RECOGNITION WORD (EDIT 25)
1129 002132  010562  000132                    MOV R5,CCT(R2)          ;AND SAVE FOR USE BY TRANSFER ROUTINES
1130                                    ;SET UP THE CONTROL CHARS FOR DP11
1131 002136  116567  000016  177646    SETSYN: MOVB CSYN(R5),C.BIMS+6  ;SYN CHAR IN SELECTED CODE
1132 002144  116562  000030  000126            MOVB BTC1(R5),BCC(R2)   ;SET BCC LATCH COUNT
```

```
1133 002152  016562  000052  000134          MOV CAKO(R5),CAK(R2)     ;ACK RECOGNITION WORD
1134 002160  005700                          TST     R0               ;WAS THE OLD CCT POINTER ZERO? (EDIT 25)
1135 002162  001405                          BEQ     SETETB           ;YES, ALL OK AS IS (EDIT 25)
1136 002164  020160  000052                  CMP     R1,CAKO(R0)      ;WAS OLD ACK REC. WORD SWAPPED? (EDIT 25)
1137 002170  001402                          BEQ     SETETB           ;NO, OK AS IS (EDIT 25)
1138 002172  000362  000134                  SWAB    CAK(R2)          ;IT WAS, SO SWAP THE NEW REC. WORD (EDIT 25)
1139 002176  116562  000005  000121  SETETB: MOVB CETB(R5),ECH(R2)
1140 002204  016562  000032  000074          MOV SYP(R5),T1A(R2)      ;SET UP FIRST XMIT PAIR FOR SYNS
1141 002212  012762  000004  000076          MOV #4,T1C(R2)
1142 002220  105062  000136                  CLRB AKT(R2)             ;INITIALIZE THE ACK COUNTER
1143                                  ; NOW WE CAN GO DO THE MODE AND MODEM CALLS TO THE ISR
1144                                  ; FIRST, EXECUTE A MODEM CONTROL CALL FOR 'TERMINAL READY'
1145 002224  012767  100003  177554          MOV #100003,C.BIMS+2             ;STORE MODEM CONTROL WORD IN ISR
1146                                                                           ;MODE CONTROL PARAMETER LIST.
1147 002232  012700  000001                  MOV #C.BMMC,R0           ;SET UP ISR MODEM SENSE
1148                                                                   ;OPCODE FOR PROPER
1149                                                                   ;SELECTION THRU MATRIX SWITCH
1150 002236  012701  002004'                 MOV #C.BIMS,R1           ;SET UP MODEM CONTROL
1151                                                                   ;PARAMETER POINTER
1152 002242  004767  0000006                 JSR PC,C.SARP            ;GO TO ISR MODE CONTROL/SENSE
1153                                                                   ;ROUTINE
1154 002246  016746  177536                  MOV C.BIMS+4,-(SP)               ;SAVE MODEM STATUS ON STACK
1155 002252  032716  000010                  BIT #10,@SP                      ;BUT MAKE SURE TERMINAL WAS READY
1156 002256  001002                          BNE PTMODE                       ;OK. PROCEED
1157 002260  005726                          TST     (SP)+
1158 002262  000472                          BR      PTEXN
1159                                  ; EXECUTE A MODE SENSE/CONTROL TO ISR
1160 002264  016567  000050  177514  PTMODE  MOV MDC(R5),C.BIMS+2      ;STORE MODE CONTROL WORD IN ISR
1161                                                                           ;MODE CONTROL PARAMETER LIST.
1162 002272  012700  000002                  MOV #C.BMDC,R0           ;SET UP ISR MODE SENSE
1163                                                                   ;OPCODE FOR PROPER
1164                                                                   ;SELECTION THRU MATRIX SWITCH
1165 002276  012701  002004'                 MOV #C.BIMS,R1           ;SET UP MODE CONTROL
1166                                                                   ;PARAMETER POINTER
1167 002302  004767  0000006                 JSR PC,C.SARP            ;GO TO ISR MODE CONTROL/SENSE
1168                                                                   ;ROUTINE
1169 002306  012600                          MOV (SP)+,R0             ;POP THE MODEM STATUS
1170 002310  032762  000100  000006          BIT #00F,LCW(R2)         ;IS THIS CALL ONLY SETTING OPTIONS?
1171 002316  001403                          BEQ PTCON                ;NO - CONTINUE THE INIT
1172 002320  016205  000006                  MOV LCW(R2),R5           ;YES - GO FINISH UP
1173 002324  000470                          BR TCJ1
```

```
1175                                        ;PROCEED WITH THE INITIALIZATION - CARRIER IS UP
1176 002326  105762  000142          PTCON: TSTB TCW(R2)              ;MASTER OR SLAVE SELECTION?
1177 002332  100036                        BPL PTSLVE                ;SLAVE
1178 002334  032762  010000  000010        BIT #XAC,LSW(R2)          ;MASTER - WAS THERE A PUTD?
1179 002342  001001                        BNE PTC3                  ;YES
1180                                        .IF NDF RSTS
1181 002344  000441                        BR      PTEXN             ;NO - MASTER INIT ERROR
1182                                        .ENDC
1183
1184                                        .IF DF  RSTS
1185                                        MOV     (SP)+,R1          ;POP R1 OFF THE STACK - EDIT 15
1186                                        BR      PTEXT
1187                                        .ENDC
1188 002346  052762  000020  000006  PTC3:  BIS #MSS,LCW(R2)           ;SET MASTER INDICATOR
1189 002354  112762  000010  000140        MOVB #ENQCNT,EKT(R2)              ; SET ENQ COUNTER
1190 002362  012700  000000                MOV #C.BTSC,R0            ;YES - SEND THE BID MESSAGE
1191 002366  016201  000130                MOV TCP(R2),R1
1192 002372  004767  000000G               JSR PC,C.SARP             ;CALL ISR VIA MATRIX SWITCH
1193 002376  004767  003676                JSR PC,ENQSET             ;SET TIMEOUT FOR BID MESSAGE RESPONSE
1194 002402  004767  000262'               JSR     PC,CSAT           ;ENABLE CRB TIMER IF NOT ALREADY ON
1195 002406                          PTEX1:
1196 002406  012601                  PTEX3: MOV (SP)+,R1              ;POP POINTER TO USER PARAMETERS
1197
1198                                        ;SET UP MODE SENSE STATUS WORD FOR USER
1199 002410  016200  000142                MOV TCW(R2),R0                    ;FETCH TERMINAL STATUS
1200 002414  042700  000525                BIC #525,R0                       ;CLEAR THE CONTROL BITS
1201 002420  010061  000004                MOV R0,4(R1)              ;AND MOVE TO USER'S STORAGE
1202 002424  000250                  PTEXT: CLN
1203
1204 002426  000412                        BR      PTEXIT
1205                                 ; SLAVE INITIALIZATION
1206 002430  032762  000001  000010  PTSLVE: BIT #RBA,LSW(R2)                ;RCV BUFFER ASSIGNED?
1207 002436  001404                        BEQ     PTEXN             ;NO-SLAVE INIT ERROR
1208 002440  042762  000020  000006  PTS1:  BIC #MSS,LCW(R2)
1209 002446  000757                        BR PTEX1                  ;CHECK FOR STATUS REQUEST
1210
1211 002450  012601                  PTEXN: MOV     (SP)+,R1          ;POP POINTER TO USER PARAMETERS
1212 002452  000270                        SEN                       ;REJECT PUTTC
1213 002454  000207                  PTEXIT: RTS    PC
```

```
                                        ; PUTTC - MESSAGE TRANSFER OR SENSE COMES HERE
                                        ;
002456  032704  100000          PTMORS: BIT #100000,R4          ;WHICH IS IT?
002462  001447                  PTM1    BEQ PTM1                ;STATUS/OPTION-SETTING
                                        ; PUTTC - MESSAGE TRANSFER ENABLE
002464  052705  000400                  BIS #INI,R5
002470  042704  037000                  BIC #37000,R4           ;CLEAR UNWANTED BITS
002474  042762  000777  000142          BIC #777,TCW(R2)                ;PREPARE TCW FOR NEW SETTINGS
002502  050462  000142                  BIS R4,TCW(R2)          ;SAVE FOR FUTURE REFERENCE
                                        ;OTHER TEST-AND-SET ACTIVITY IS HANDLED IN THE SAME WAY
                                        ; NATIVE MODE?
002506  032704  000020          TCJ1:   BIT #20,R4
002512  001407                          BEQ TCJ2                ;NOT REQUESTED
002514  042705  001000                  BIC #NAT,R5
002520  032704  000010                  BIT #10,R4
002524  001402                          BEQ TCJ2
002526  052705  001000                  BIS #NAT,R5
                                        ; TRANSPARENCY ACTION?
002532  032704  000004          TCJ2:   BIT #4,R4
002536  001407                          BEQ TCJ3                ;NONE REQUESTED
002540  042705  020000                  BIC #XPT,R5
002544  032704  000002                  BIT #2,R4               ;SET FLAG TO DESIRED STATE
002550  001402                          BEQ TCJ3
002552  052705  020000                  BIS #XPT,R5             ;NEXT NON-ITB PUTD WILL TOGGLE IT
                                        ; SET MASTER SLAVE SELECT FLAG
002556  042705  000020          TCJ3:   BIC #MSS,R5
002562  105762  000142                  TSTB TCW(R2)
002566  100002                          BPL PTM2
002570  052705  000020                  BIS #MSS,R5
002574  010562  000006          PTM2:   MOV R5,LCW(R2)          ;RESTORE LCW TO LINE TABLE
002600  000702                          BR      PTEX3
                                        ;
                                        ;STATUS AND OPTION SETTING CALL
002602  052705  000100          PTM1:   BIS #00F,R5             ;SET FLAG TO INDICATE OPTIONS ONLY
002606  042762  007777  000142          BIC #7777,TCW(R2)       ;CLEAR UNWANTED BITS
002614  000167  177262                  JMP PTC4
```

```
                                        ; PUTTC - TERMINATION
002620  012700  000001          PTTERM: MOV #C.BMMC,R0          ;RESET TERMINAL READY
002624  012767  100001  177154          MOV #100001,C.BIMS+2
002632  012701  002004'                 MOV #C.BIMS,R1
002636  004767  000000G                 JSR PC,C.SARP
                                        .IF NDF RSTS    ;IF ASSEMBLED FOR RSTS LEAVE THIS ROUTINE OUT
002642  032762  000001  000010  PTT1:   BIT #RBA,LSW(R2)              ;RELEASE RECEIVE BUFFERS IF AN.
002650  001403                         BEQ PTTEXT
002652  004767  175246                 JSR PC,C.BRCD
002656  000771                         BR PTT1                 ;JUST IN CASE THERE WERE 2 BUFFERS
                                        .ENDC           ;END RSTS

002660  005062  000010          PTTEXT: CLR LSW(R2)              ;RETURN TO CONTROL MODE
002664  005062  000006                  CLR LCW(R2)
002670  005062  000012                  CLR RSW(R2)
002674  005062  000014                  CLR TSW(R2)
002700  012601                         MOV     (SP)+,R1        ;POP POINTER TO USER PARAMETERS
002702  000207                         RTS     PC



                                        .IF NDF RSX
                                        .EOT
                                        ;;;;;;;;;;;;;;;;;;;;
                                        ;END OF TAPE 2
                                        .ENDC
                                        ;BISYNC TAPE 3 OF 5
                                        ;-ASRBUF EMT COMMAND SERVICE ROUTINE
                                        ;ENTERED FROM TERMINAL TYPE MATRIX SWITCH WITH:
                                        ;       FUNCTION CODE   003     - ASSIGN ONLY
                                        ;                       043     - ASSIGN + ACK
                                        ;                       103     - ASSIGN + NAK

                                        ;R1:    POINTER TO ASRBUF PARAMETER LIST
                                        ;R2:    POINTER TO FIRST WORD OF LINE TABLE
                                        ;R3:    LINE#*2
                                        ; R0,R1,R4 AND R5 ARE DESTROYED BY THIS ROUTINE

                                        ;FORMAT OF ASRBUF PARAMETER LIST:
                                        ;LOCP:          STARTING ADDRESS OF RECEIVE BUFFER
                                        ;LOCP+2:        MAXIMUM BUFFER SIZE IN BYTES

002704  105721                  C.BARB: TSTB    (R1)+           ;POINT TO SECOND BYTE IN PARAM. LIST
002706  112100                          MOVB    (R1)+,R0                ;GET ASRBUF OP CODE.
002710  012105                          MOV     (R1)+,R5                ;BUFFER ADDRESS
002712  011104                          MOV     (R1),R4                 ;BYTE COUNT
002714  005704                  C.BAR1: TST     R4                      ;COUNT POSITIVE NON-ZERO?
002716  003417                          BLE     C.BARX          ;NO - REJECT
002720  016201  000016                  MOV     ASX(R2),R1              ;LOCATE CURRENT ASSIGN SLOT
002724  060201                          ADD     R2,R1
002726  062701  000026                  ADD     #UBL,R1
                                        .IF DF LSI11
                                        MFPS    -(SP)           ;PREVENT INTERFERENCE...
                                        MTPS#340                ;...BY RECEIVE DONE CODE
                                        .IFF
002732  013746  177776                  MOV     @#C.STAT,-(SP)          ;PREVENT INTERFERENCE...
002736  112737  000340  177776          MOVB    #340,@#C.STAT           ;...BY RECEIVE DONE CODE
```

```
                                        .ENDC
                                        TST     (R1)                    ;THIS SLOT AVAILABLE?
                                        BEQ     C.BAB1
                                ; NO SLOT AVAILABLE - REJECT THE ASRBUF
                                        .IF DF LSI11
                                        MTPS    (SP)+           ;RESTORE PRIORITY
                                        .IFF
                                        MOV     (SP)+,@#C.STAT          ;RESTORE PRIORITY
                                        .ENDC
                                        CLV
                                C.BARX: SEN                             ;SET FAIL INDICATION
                                        BR      ASSGN3

                                ; ASSIGN THE INCOMING BUFFER TO THE CURRENT ASSIGN SLOT
                                C.BAB1: MOV     R5,(R1)+                ;MOVE PARAMETERS
                                        MOV     R4,(R1)
                                        ADD     #4,ASX(R2)              ;UPDATE THE SLOT INDEX
                                        CMP     ASX(R2),#MBX            ;NEEDS RESETTING?
                                        BLT     ASSGN1                  ;NO
                                        CLR     ASX(R2)                 ;YES - CLEAR IT
                                ASSGN1: BIT     #RBA,LSW(R2)            ;IS THERE ALREADY AN ACTIVE BUFFER?
                                        BNE     ASSGN2                  ;YES - WE'RE DONE
                                        MOV     R5,ARA(R2)              ;NO - M KE THIS ONE IT
                                        .IF DF RSTS
                                        JSR     PC,GTBF1                ;GET MAPPED ADDRESS FOR BUFFER
                                        .ENDC
                                        MOV     R4,ABC(R2)
                                        BIS     #RBA,LSW(R2)
                                ASSGN2: .IF DF LSI11
                                        MTPS    (SP)+           ;RESTORE ORIGIANL PRIORITY
                                        .IFF
                                        MOV     (SP)+,@#C.STAT          ;RESTORE ORIGINAL PRIORITY
                                        .ENDC
                                        JSR     PC,ARBSP                ;GO SEE IF ACK/NAK NEEDED
                                        CLNV                            ;SUCCESS INDICATION
                                ASSGN3: RTS     PC
                                ; TEST IF ASRBUF REQUEST INCLUDES ACK/NAK ACTIVITY
                                        .IF DF RSX
                                        .PSECT ISR
                                        .ENDC
                                ARBSP:  MOV CCT(R2),R1          ;POINTER TO CODE CONTROL TABLE
                                        BIT #140,R0             ;IS IT ACK/NAK REQUEST
                                        BEQ ARRCV               ;NO
                                ; WHICH TYPE OF REQUEST WAS IT?
                                        BIT #40,R0              ;IS IT ACK TYPE REQUEST (43)
                                        BEQ TRYNAK              ;NO - MUST BE NAK TYPE
                                        MOV AMP(R1),T2A(R2)     ;YES - SET UP THE ACK MESSAGE
                                        TSTB AKT(R2)
                                        BEQ ARB1
                                        ADD #3,T2A(R2)
                                ARB1:   MOV #3,T2C(R2)
                                        COMB AKT(R2)            ;TOGGLE ACK COUNTER
                                ARBSET: MOV R1, T3A(R2)         ;SET UP EXTRA PAD
                                        ADD #CPAD,T3A(R2)
                                        MOV #1,T3C(R2)
                                ACKOUT: MOV TCP(R2),R1          ;SET XMIT POINTER
                                        MOV #C.BTSC,R0          ;MATRIX SWITCH CODE
```

```
1366 000160  004767  000000G                    JSR PC,C.SARP          ;ISR OUTPUT STARTUP
1367 000164  052762  000004  000010             BIS     #AAF,LSW(R2)   ;FLAG ACK AVAILABLE IN CASE OF ENQ
1368                                                                    ;FROM OTHER SIDE
1369 000172  000207                    ARSEXT:  RTS PC                 ;GO BACK TO TAKE EMT EXIT
1370
1371                                    ;NAK TYPE REQUEST (103)
1372 000174  016162  000044  000100    TRYNAK:  MOV NMP(R1),T2A(R2)    ;YES - SET POINTER TO NAK MESSAGE
1373 000202  012762  000002  000102             MOV #2,T2C(R2)
1374 000210  000747                             BR ARBSET
1375
1376
1377                                    ; NO ACK/NAK - DO THE RECEIVE ENABLE
1378 000212  032762  002000  000010    ARRCV:   BIT #ITBA,LSW(R2)      ;IF ITBA STILL ACTIVE...
1379 000220  001364                             BNE ARSEXT             ;DON'T DO TURNAROUND
1380 000222  105762  000127                     TSTB BCL(R2)           ;DITTO IF STILL IN BCC PHASE
1381 000226  001361                             BNE ARSEXT
1382 000230                            ARRCV1:
1383 000230  005067  002004'                    CLR     C.BIMS         ;PUT LINE NUMBER IN PARAM LIST
1384 000234  012700  000002                     MOV #C.BMDC,R0         ;CODE FOR MODE CONTROL CALL
1385 000240  016201  000132                     MOV     CCT(R2),R1     ;GET CODE TABLE POINTER
1386 000244  016167  000050  002006'            MOV MDC(R1),C.BIMS+2   ;MODE WORD
1387 000252  012701  002004'                    MOV #C.BIMS,R1
1388 000256  004767  000000G                    JSR PC,C.SARP
1389 000262                            CSAT:
1390                                            .IF Z   RCVSIB
1391 000262  032762  000002  000010             BIT #CSA,LSW(R2)       ;CRB SERVICE ALREADY ACTIVE?
1392 000270  001340                             BNE ARSEXT             ;YES. WE'RE DONE
1393 000272  052762  000002  000010             BIS #CSA,LSW(R2)       ;NO. SET IT UP NOW
1394 000300  005000                             CLR     R0             ;PUT LINE NUMBER IN R0
1395 000302  116204  000120                     MOVB TCT(R2),R4
1396 000306  012705  004022'                    MOV #CRBSRV,R5
1397 000312  000167  000000G                    JMP     CTIME
1398                                            .ENDC
1399                                            .IF NZ  RCVSIB
1400                                            BR      ARSEXT
1401                                            .ENDC
1402
1403
```

```
1405                                    ; TRANSMIT DONE SIB ENTRY PROCESSOR
1406                                    ;;ENTERED FROM BUFFER UNLOADER WITH:
1407                                    ;R0 CONTAINING LINE# AND TYPE CODE
1408                                    ;R2 CONTAINING POINTER TO BASE OF LINE TABLE.
1409                                    ;R3 CONTAINING LINE# AS A WORD INDEX.
1410
1411
1412 000316  032762  001000  000010  C.BODE: BIT     #RDBF,LSW(R2)           ;HAVE DATA MESSAGE POINTERS
1413 000324  001414                          BEQ     C.BOD1                  ;BEEN SAVED?
1414                                         ;YES -RESTORE OLD POINTERS
1415 000326  016262  000144  000100          MOV     TT2A(R2),T2A(R2)
1416 000334  016262  000146  000102          MOV     TT2C(R2),T2C(R2)
1417 000342  016262  000150  000104          MOV     TT3A(R2),T3A(R2)
1418 000350  042762  001000  000010          BIC     #RDBF,LSW(R2)
1419 000356  032762  000400  000006  C.BOD1: BIT     #INI,LCW(R2)            ;IF SLAVE INITIALIZATION...
1420 000364  001007                         BNE C.B1                        ;ALWAYS DO XMIT DONE
1421 000366  032762  000020  000006         BIT #MSS,LCW(R2)                ;IF MASTER INIT, PROCEED NORMALLY
1422 000374  001003                         BNE C.B1
1423 000376  004767  000130          C.BO.  JSR PC,TDENT
1424 000402  000410                         BR C.BUD2
1425 000404  032762  040000  000010  C.B1:  BIT #ANF,LSW(R2)                ;IF NOT SET...
1426 000412  001404                         BEQ C.BUD2                      ; DON'T WANT AN XMIT DONE
1427 000414  032762  020000  000010         BIT #AEF,LSW(R2)                ;IF AEF SET ...
1428 000422  001765                         BEQ     C.BO            ;
1429
1430 000424  004767  177562          C.BUD2: JSR     PC,ARRCV                ;RESET LINE
1431 000430  032762  001000  000014         BIT     #ESNT,TSW(R2)           ;IF EOT WAS SENT ...
1432 000436  001412                         BEQ     C.BUD3
1433 000440  016201  000132                 MOV     CCT(R2),R1              ;GET CODE CONTROL POINTER
1434 000444  016162  000052  000134         MOV     CAK0(R1),CAK(R2)        ;RESET MASTER ACK COUNT
1435 000452  105062  000136                 CLRB    AKT(R2)                 ;CLEAR SLAVE ACK COUNT
1436 000456  042762  001420  000006         BIC     #MSS!INI!NAT,LCW(R2)    ;..CLEAR MASTER AND MESSAGE XFER
1437                                                                         ;AND NATIVE MODE
1438 000464  032762  020000  000010  C.BUD3: BIT     #AEF,LSW(R2)            ;ACK EXPECTED?
1439 000472  001407                         BEQ     C.BUD4                  ;NO
1440 000474  032762  000400  000006         BIT     #INI,LCW(R2)            ;ALSO MESSAGE TRANSFER?
1441 000502  001403                         BEQ     C.BUD4                  ;NO
1442                                    ;MASTER-MESSAGE X-MIT COMPLETE:  START RCVE TIMER
1443
1444                                         .IF DF  UAPTIM
1445 000504  116762  000002' 000115         MOVB    MSRT,MRC(R2)            ;TIMEOUT VALUE IN TIC INTERVALS
1446                                         .IFF
1447                                         MOVB    #MSRT,MRC(R2)           ;TIMEOUT VALUE
1448                                         .ENDC
1449 000512  032762  002000  000014  C.BUD4: BIT     #BSNT,TSW(R2)           ;IF BID MESSAGE, LEAVE BSNT SET
1450 000520  001003                         BNE     C.B5
1451 000522  042762  177277  000014         BIC     #177277,TSW(R2)         ;CLEAR TRANSMIT STATUS
1452                                                                         ; EXCEPT MASTER TIMEOUT - EDIT 15
1453 000530  000207                  C.B5:  RTS     PC                      ;RETURN TO BUFFER UNLOADER.
```

```
                                        ;TRANSMIT DONE UIB ENTRY GENERATOR

                                                .IF DF RSTS
                                        TDENTO: CLR     R0                      ;CLEAR LINE NUMBER
                                        TDENT:  CLR     -(SP)                   ;
                                                .IFF
1461 000532  116205  000141            TDENT:  MOVB    RKT(R2),R5              ;SAVE NUMBER OF ERRORS TO GO UNTIL
146.                                                                            ;ABORTING IN R5
1463 000536  005405                             NEG     R5                      ;AND SUBTRACT FROM STARTING VALUE
1464 000540  062705  000010                     ADD     #NAKCNT,R5              ;TO OBTAIN # OF XMIT ERRORS
1465 000544  010546                             MOV     R5,-(SP)                ;STORE IT FOR UIB ENTRY
1466                                            .ENDC
1467 000546  016246  000014                     MOV     TSW(R2),-(SP)           ;STORE TRANSMIT STATUS WORD
1468 000552  016246  000066                     MOV     TSA(R2),-(SP)           ;STORE BUFFER ADDRESS
1469 000556  010046                             MOV     R0,-(SP)                ;STORE LINE #
1470 000560  112766  000304  000001            MOVB    #C.BGTD,1(SP)           ;STORE X-MIT DONE UIB CODE
1471 000566  010605                             MOV     SP,R5                   ;STORE UIB ADDRESS POINTER
1472 000570  004767  000000G                    JSR     PC,C.SFY                ;CALL UIB LOADER
1473 000574  062706  000010                     ADD     #10,SP                  ;RESTORE STACK POINTER
1474 000600  042762  074020  000010            BIC     #AEF!ANF!XAC!NEF!STXF,LSW(R2)   ;EDIT 17
1475 000606  112762  000010  000140            MOVB    #ENQCNT,EKT(R2)         ;SET ENQ COUNTER
1476 000614  042762  176777  000014            BIC     #176777,TSW(R2)         ;DON'T CLEAR ESNT
1477 000622  000207                             RTS     PC
```

```
1479
1480 003052                            .IF DF RSX
1481                                    .PSECT  XXX
1482                                    .ENDC
1483
1484                            ; EBCDIC-TO-ASCII TRANSLATION TABLE
1485                                    .IF NZ CA.EBC
1486                            ;000
1486 003052      000            ETA:    .BYTE   000             ;NUL
1487 003053      001                    .BYTE   001             ;SOH
1488 003054      002                    .BYTE   002             ;STX
1489 003055      003                    .BYTE   003             ;ETX
1490 003056      000                    .BYTE   000             ;PF -NO ASCII
1491 003057      011                    .BYTE   011             ;HT
1492 003060      000                    .BYTE   000             ;LC - NO ASCII
1493 003061      177                    .BYTE   177             ;DEL
1494                            ;010
1495 003062      000                    .BYTE   000             ;N.A.
1496 003063      000                    .BYTE   000             ;N.A.
1497 003064      000                    .BYTE   000             ;SMM - N.A.
1498 003065      013                    .BYTE   013             ;VT
1499 003066      014                    .BYTE   014             ;FF
1500 003067      015                    .BYTE   015             ;CR
1501 003070      016                    .BYTE   016             ;SO
1502 003071      017                    .BYTE   017             ;SI
1503                            ;020
1504 003072      020                    .BYTE   020             ;DLE
1505 003073      021                    .BYTE   021             ;DC1
1506 003074      022                    .BYTE   022             ;DC2
1507 003075      023                    .BYTE   023             ;DC3(TM)
1508 003076      000                    .BYTE   000             ;RES - N.A.
1509 003077      000                    .BYTE   000             ;NL - N.A.
1510 003100      010                    .BYTE   010             ;BS
1511 003101      000                    .BYTE   000             ;IL - N.A.
1512                            ;030
1513 003102      030                    .BYTE   030             ;CAN
1514 003103      031                    .BYTE   031             ;EM
1515 003104      000                    .BYTE   000             ;CC - N.A.
1516 003105      000                    .BYTE   000             ;N.A.
1517 003106      034                    .BYTE   034             ;IFS
1518 003107      035                    .BYTE   035             ;IGS
1519 003110      036                    .BYTE   036             ;IRS
1520 003111      037                    .BYTE   037             ;IUS(ITB)
```

```
1522                                    ;040
1523 003112  000                                .BYTE   000             ;DS - N.A.
1524 003113  000                                .BYTE   000             ;SOS -N A.
1525 003114  000                                .BYTE   000             ;FS - N.A.
1526 003115  000                                .BYTE   000             ;N.A.
1527 003116  000                                .BYTE   000             ;BYP - N.A.
1528 003117  012                                .BYTE   012             ;LF
1529 003120  027                                .BYTE   027             ;ETB(EOB)
1530 003121  033                                .BYTE   033             ;ESC(PRE)
1531                                    ;050
1532 003122  000                                .BYTE   000             ;N.A.
1533 003123  000                                .BYTE   000             ;N.A.
1534 003124  000                                .BYTE   000             ;SM -N.A.
1535 003125  000                                .BYTE   000             ;N.A.
1536 003126  000                                .BYTE   000             ;N.A.
1537 003127  005                                .BYTE   005             ;ENQ
1538 003130  006                                .BYTE   006             ;ACK
1539 003131  007                                .BYTE   007             ;BEL
1540                                    ;060
1541 003132  000                                .BYTE   000             ;N.A.
1542 003133  000                                .BYTE   000             ;N.A.
1543 003134  026                                .BYTE   026             ;SYN
1544 003135  000                                .BYTE   000             ;N.A.
1545 003136  000                                .BYTE   000             ;PN - N.A.
1546 003137  000                                .BYTE   000             ;RS - N.A.
1547 003140  000                                .BYTE   000             ;UC - N.A.
1548 003141  004                                .BYTE   004             ;EOT
1549                                    ;070
1550 003142  000                                .BYTE   000             ;N.A.
1551 003143  000                                .BYTE   000             ;N.A.
1552 003144  000                                .BYTE   000             ;N.A.
1553 003145  000                                .BYTE   000             ;N.A.
1554 003146  024                                .BYTE   024             ;DC4
1555 003147  025                                .BYTE   025             ;NAK
1556 003150  000                                .BYTE   000             ;N.A.
1557 003151  032                                .BYTE   032             ;SUB
```

```
1559                                    ;100
1560 003152  040                               .BYTE   040             ;SPACE
1561 003153  000                               .BYTE   000             ;N.A.
1562 003154  000                               .BYTE   000             ;N.A.
1563 003155  000                               .BYTE   000             ;N.A.
1564 003156  000                               .BYTE   000             ;N.A.
1565 003157  000                               .BYTE   000             ;N.A.
1566 003160  000                               .BYTE   000             ;N.A.
1567 003161  000                               .BYTE   000             ;N.A.
1568                                    ;110
1569 003162  000                               .BYTE   000             ;N.A.
1570 003163  000                               .BYTE   000             ;N.A.
1571 003164  133                               .BYTE   133             ;CENT SIGN (  IN ASCII)
1572 003165  056                               .BYTE   056             ;.
1573 003166  074                               .BYTE   074             ;<
1574 003167  050                               .BYTE   050             ;(
1575 003170  053                               .BYTE   053             ;+
1576 003171  136                               .BYTE   136             ;VERTICAL BAR
1577                                    ;120
1578 003172  046                               .BYTE   046             ;&
1579 003173  000                               .BYTE   000             ;N.A.
1580 003174  000                               .BYTE   000             ;N.A.
1581 003175  000                               .BYTE   000             ;N.A.
1582 003176  000                               .BYTE   000             ;N.A.
1583 003177  000                               .BYTE   000             ;N.A.
1584 003200  000                               .BYTE   000             ;N.A.
1585 003201  000                               .BYTE   000             ;N.A.
1586                                    ;130
1587 003202  000                               .BYTE   000             ;N.A.
1588 003203  000                               .BYTE   000             ;N.A.
1589 003204  041                               .BYTE   041             ;!
1590 003205  044                               .BYTE   044             ;$
1591 003206  052                               .BYTE   052             ;*
1592 003207  051                               .BYTE   051             ;)
1593 003210  073                               .BYTE   073             ;;
1594 003211  135                               .BYTE   135             ;  (WAS EBCDIC "NOT" OPERATOR)
```

```
1596                                    ;140
1597 003212     055                             .BYTE   055             ;-
1598 003213     057                             .BYTE   057             ;/
1599 003214     000                             .BYTE   000             ;N.A.
1600 003215     000                             .BYTE   000             ;N.A.
1601 003216     000                             .BYTE   000             ;N.A.
1602 003217     000                             .BYTE   000             ;N.A.
1603 003220     000                             .BYTE   000             ;N.A.
1604 003221     000                             .BYTE   000             ;N.A.
1605                                    ;150
1606 003222     000                             .BYTE   000             ;N.A.
1607 003223     000                             .BYTE   000             ;N.A.
1608 003224     174                             .BYTE   174             ;NON-PRINTING (ASCII - |)
1609 003225     054                             .BYTE   054             ;,
1610 003226     045                             .BYTE   045             ;%
1611 003227     137                             .BYTE   137             ;_
1612 003230     076                             .BYTE   076             ;>
1613 003231     077                             .BYTE   077             ;?
1614                                    ;160
1615 003232     000                             .BYTE   000             ;N.A.
1616 003233     000                             .BYTE   000             ;N.A.
1617 003234     000                             .BYTE   000             ;N.A.
1618 003235     000                             .BYTE   000             ;N.A.
1619 003236     000                             .BYTE   000             ;N.A.
1620 003237     000                             .BYTE   000             ;N.A.
1621 003240     000                             .BYTE   000             ;N.A.
1622 003241     000                             .BYTE   000             ;N.A.
1623                                    ;170
1624 003242     000                             .BYTE   000             ;N.A.
1625 003243     140                             .BYTE   140             ;NON-PRINTING
1626 003244     072                             .BYTE   072             ;:
1627 003245     043                             .BYTE   043             ;#
1628 003246     100                             .BYTE   100             ;@
1629 003247     047                             .BYTE   047             ;'
1630 003250     075                             .BYTE   075             ;=
1631 003251     042                             .BYTE   042             ;"
```

```
1633                                    ;200
1634 003252  000                        .BYTE   000             ;N.A.
1635 003253  141                        .BYTE   141             ;A - LOWER CASE
1636 003254  142                        .BYTE   142             ;B
1637 003255  143                        .BYTE   143             ;C
1638 003256  144                        .BYTE   144             ;D
1639 003257  145                        .BYTE   145             ;E
1640 003260  146                        .BYTE   146             ;F
1641 003261  147                        .BYTE   147             ;G
1642                                    ;210
1643 003262  150                        .BYTE   150             ;H - LOWER CASE
1644 003263  151                        .BYTE   151             ;I
1645 003264  000                        .BYTE   000             ;N.A.
1646 003265  000                        .BYTE   000             ;N.A.
1647 003266  000                        .BYTE   000             ;N.A.
1648 003267  000                        .BYTE   000             ;N.A.
1649 003270  000                        .BYTE   000             ;N.A.
1650 003271  000                        .BYTE   000             ;N.A.
1651                                    ;220
1652 003272  000                        .BYTE   000             ;N.A.
1653 003273  152                        .BYTE   152             ;J - LOWER CASE
1654 003274  153                        .BYTE   153             ;K
1655 003275  154                        .BYTE   154             ;L
1656 003276  155                        .BYTE   155             ;M
1657 003277  156                        .BYTE   156             ;N
1658 003300  157                        .BYTE   157             ;O
1659 003301  160                        .BYTE   160             ;P
1660                                    ;230
1661 003302  161                        .BYTE   161             ;Q - LOWER CASE
1662 003303  162                        .BYTE   162             ;R
1663 003304  000                        .BYTE   000             ;N.A.
1664 003305  000                        .BYTE   000             ;N.A.
1665 003306  000                        .BYTE   000             ;N.A.
1666 003307  000                        .BYTE   000             ;N.A.
1667 003310  000                        .BYTE   000             ;N.A.
1668 003311  000                        .BYTE   000             ;N.A.
```

```
1670                                    ;240
1671 003312      000                            .BYTE   000             ;N.A.
1672 003313      176                            .BYTE   176             ;NON-PRINTING
1673 003314      163                            .BYTE   163             ;S
1674 003315      164                            .BYTE   164             ;T
1675 003316      165                            .BYTE   165             ;U
1676 003317      166                            .BYTE   166             ;V
1677 003320      167                            .BYTE   167             ;W
1678 003321      170                            .BYTE   170             ;X
1679                                    ;250
1680 003322      171                            .BYTE   171             ;Y - LOWER CASE
1681 003323      172                            .BYTE   172             ;Z
1682 003324      000                            .BYTE   000             ;N.A.
1683 003325      000                            .BYTE   000             ;N.A.
1684 003326      000                            .BYTE   000             ;N.A.
1685 003327      000                            .BYTE   000             ;N.A.
1686 003330      000                            .BYTE   000             ;N.A.
1687 003331      000                            .BYTE   000             ;N.A.
1688                                    ;260
1689 003332      000                            .BYTE   000             ;N.A.
1690 003333      000                            .BYTE   000             ;N.A.
1691 003334      000                            .BYTE   000             ;N.A.
1692 003335      000                            .BYTE   000             ;N.A.
1693 003336      000                            .BYTE   000             ;N.A.
1694 003337      000                            .BYTE   000             ;N.A.
1695 003340      000                            .BYTE   000             ;N.A.
1696 003341      000                            .BYTE   000             ;N.A.
1697                                    ;270
1698 003342      000                            .BYTE   000             ;N.A.
1699 003343      000                            .BYTE   000             ;N.A.
1700 003344      000                            .BYTE   000             ;N.A.
1701 003345      000                            .BYTE   000             ;N.A.
1702 003346      000                            .BYTE   000             ;N.A
1703 003347      000                            .BYTE   000             ;N.A.
1704 003350      000                            .BYTE   000             ;N.A.
1705 003351      000                            .BYTE   000             ;N.A.
```

```
1707                                    ;300
1708 003352      173                            .BYTE   173             ;NON-PRINTING
1709 003353      101                            .BYTE   101             ;A
1710 003354      102                            .BYTE   102             ;B
1711 003355      103                            .BYTE   103             ;C
1712 003356      104                            .BYTE   104             ;D
1713 003357      105                            .BYTE   105             ;E
1714 003360      106                            .BYTE   106             ;F
1715 003361      107                            .BYTE   107             ;G
1716                                    ;310
1717 003362      110                            .BYTE   110             ;H
1718 003363      111                            .BYTE   111             ;I
1719 003364      000                            .BYTE   000             ;N.A.
1720 003365      000                            .BYTE   000             ;N.A.
1721 003366      000                            .BYTE   000             ;N.A.
1722 003367      000                            .BYTE   000             ;N.A.
1723 003370      000                            .BYTE   000             ;N.A.
1724 003371      000                            .BYTE   000             ;N.A.
1725                                    ;320
1726 003372      175                            .BYTE   175             ;NON-PRINTING
1727 003373      112                            .BYTE   112             ;J
1728 003374      113                            .BYTE   113             ;K
1729 003375      114                            .BYTE   114             ;L
1730 003376      115                            .BYTE   115             ;M
1731 003377      116                            .BYTE   116             ;N
1732 003400      117                            .BYTE   117             ;O
1733 003401      120                            .BYTE   120             ;P
1734                                    ;330
1735 003402      121                            .BYTE   121             ;Q
1736 003403      122                            .BYTE   122             ;R
1737 003404      000                            .BYTE   000             ;N.A.
1738 003405      000                            .BYTE   000             ;N.A.
1739 003406      000                            .BYTE   000             ;N.A.
1740 003407      000                            .BYTE   000             ;N.A.
1741 003410      000                            .BYTE   000             ;N.A.
1742 003411      000                            .BYTE   000             ;N.A.
```

```
1744                                    ;340
1745 003412     134                             .BYTE   134             ;
1746 003413     000                             .BYTE   000             ;N.A.
1747 003414     123                             .BYTE   123             ;S
1748 003415     124                             .BYTE   124             ;T
1749 003416     125                             .BYTE   125             ;U
1750 003417     126                             .BYTE   126             ;V
1751 003420     127                             .BYTE   127             ;W
1752 003421     130                             .BYTE   130             ;X
1753                                    ;350
1754 003422     131                             .BYTE   131             ;Y
1755 003423     132                             .BYTE   132             ;Z
1756 003424     000                             .BYTE   000             ;N.A.
1757 003425     000                             .BYTE   000             ;N.A.
1758 003426     000                             .BYTE   000             ;N.A.
1759 003427     000                             .BYTE   000             ;N.A.
1760 003430     000                             .BYTE   000             ;N.A.
1761 003431     000                             .BYTE   000             ;N.A.
1762                                    ;360
1763 003432     060                             .BYTE   060             ;0
1764 003433     061                             .BYTE   061             ;1
1765 003434     062                             .BYTE   062             ;2
1766 003435     063                             .BYTE   063             ;3
1767 003436     064                             .BYTE   064             ;4
1768 003437     065                             .BYTE   065             ;5
1769 003440     066                             .BYTE   066             ;6
1770 003441     067                             .BYTE   067             ;7
1771                                    ;370
1772 003442     070                              BYTE   070             ;8
1773 003443     071                             .BYTE   071             ;9
1774 003444     000                             .BYTE   000             ;N.A.
1775 003445     000                             .BYTE   000             ;N.A.
1776 003446     000                             .BYTE   000             ;N.A.
1777 003447     000                             .BYTE   000             ;N.A.
1778 003450     000                             .BYTE   000             ;N.A.
1779 003451     177                             .BYTE   177             ;N.A.  - BUT RECOGNIZE AS PAD
```

```
1781
1782                                            .IF DF  ASCI
1783                                    ;
1784                                    ;
1785
1786                                            .MACRO  HEX     C3,C4
1787                                    C1=''C3-60
1788                                    C2=''C4-60
1789                                    .IF   GT C1-9.    ;SUBT 7 MORE IF A LETTER
1790                                    C1=C1-7
1791                                    .ENDC
1792                                    .IF   GT C2-9.
1793                                    C2=C2-7
1794                                    .ENDC
1795
1796                                            .BYTE   C1*16.+C2
1797                                            .ENDM
1798
1799                                    ;
1800                                    ;
1801 003452                            AEXB:
1802                                    ;
1803                                    ; 000
1804 003452                                     HEX     0,0     ;NULL
1805 003453                                     HEX     0,1     ;SOH
1806 003454                                     HEX     0,2     ;STX
1807 003455                                     HEX     0,3     ;ETX
1808 003456                                     HEX     3,7     ;EOT
1809 003457                                     HEX     2,D     ;ENQ
1810 003460                                     HEX     2,E     ;ACK
1811 003461                                     HEX     2,F     ;BEL
1812                                    ;
1813                                    ; 010
1814 003462                                     HEX     1,6     ;BS
1815 003463                                     HEX     0,5     ;HT
1816 003464                                     HEX     2,5     ;LF
1817 003465                                     HEX     0,B     ;VT
1818 003466                                     HEX     0,C     ;FF
1819 003467                                     HEX     0,D     ;CR
1820 003470                                     HEX     0,E     ;SO
1821 003471                                     HEX     0,F     ;SI
1822                                    ;
1823                                    ; 020
1824 003472                                     HEX     1,0     ;DLE
1825 003473                                     HEX     1,1     ;DC1
1826 003474                                     HEX     1,2     ;DC2
1827 003475                                     HEX     1,3     ;DC3
1828
1829 003476                                     HEX     3,C     ;DC4
1830 003477                                     HEX     3,D     ;NAK
1831 003500                                     HEX     3,2     ;SYN
1832 003501                                     HEX     2,6     ;ETB
1833                                    ;
1834                                    ; 030
1835 003502                                     HEX     1,8     ;CAN
1836 003503                                     HEX     1,9     ;EM
1837 003504                                     HEX     3,F     ;SUB
```

```
1838 003505                                     HEX     2,7     ;ESC
1839 003506                                     HEX     2,2     ;FS
1840 003507                                     HEX     1,0     ;ASCII GS BECOMES EBCDIC IGS
1841 003510                                     HEX     3,5     ;RS
1842 003511                                     HEX     1,F     ;ASCII US BECOMES EBCDIC IUS
1843                                    ;
1844                                    ; 040
1845 003512                                     HEX     4,0     ;SPACE
1846 003513                                     HEX     5,A     ;!
1847 003514                                     HEX     7,F     ;"
1848 003515                                     HEX     7,B     ;#
1849 003516                                     HEX     5,B     ;$
1850 003517                                     HEX     6,C     ;%
1851 003520                                     HEX     5,0     ;&
1852 003521                                     HEX     7,D     ;'
1853                                    ;
1854                                    ; 050
1855 003522                                     HEX     4,D     ;(
1856 003523                                     HEX     5,D     ;)
1857 003524                                     HEX     5,C     ;*
1858 003525                                     HEX     4,E     ;+
1859 003526                                     HEX     6,B     ;,
1860 003527                                     HEX     6,0     ;-
1861 003530                                     HEX     4,B     ;.
1862 003531                                     HEX     6,1     ;/
1863                                    ;
1864                                    ; 060
1865 003532                                     HEX     F,0     ;0
1866 003533                                     HEX     F,1     ;1
1867 003534                                     HEX     F,2     ;2
1868 003535                                     HEX     F,3     ;3
1869 003536                                     HEX     F,4     ;4
1870 003537                                     HEX     F,5     ;5
1871 003540                                     HEX     F,6     ;6
1872 003541                                     HEX     F,7     ;7
1873                                    ;
1874                                    ; 070
1875 003542                                     HEX     F,8     ;8
1876 003543                                     HEX     F,9     ;9
1877 003544                                     HEX     7,A     ;:
1878 003545                                     HEX     5,E     ;;
1879 003546                                     HEX     4,C     ;<
1880 003547                                     HEX     7,E     ;=
1881 003550                                     HEX     6,E     ;>
1882 003551                                     HEX     6,F     ;?
1883                                    ;
1884                                    ;       LOWER CASE
1885                                    ;
1886                                    ; 100
1887 003552                                     HEX     7,C     ;@
1888 003553                                     HEX     C,1     ;A
1889 003554                                     HEX     C,2     ;B
1890 003555                                     HEX     C,3     ;C
1891 003556                                     HEX     C,4     ;D
1892 003557                                     HEX     C,5     ;E
1893 003560                                     HEX     C,6     ;F
1894 003561                                     HEX     C,7     ;G
```

```
1895                                    ; 110
1896 003562                                     HEX     C,8     ;H
1897 003563                                     HEX     C,9     ;I
1898 003564                                     HEX     D,1     ;J
1899 003565                                     HEX     D,2     ;K
1900 003566                                     HEX     D,3     ;L
1901 003567                                     HEX     D,4     ;M
1902 003570                                     HEX     D,5     ;N
1903 003571                                     HEX     D,6     ;O
1904                                    ;
1905                                    ; 120
1906 003572                                     HEX     D,7     ;P
1907 003573                                     HEX     D,8     ;Q
1908 003574                                     HEX     D,9     ;R
1909 003575                                     HEX     E,2     ;S
1910 003576                                     HEX     E,3     ;T
1911 003577                                     HEX     E,4     ;U
1912 003600                                     HEX     E,5     ;V
1913 003601                                     HEX     E,6     ;W
1914                                    ;
1915                                    ; 130
1916 003602                                     HEX     E,7     ;X
1917 003603                                     HEX     E,8     ;Y
1918 003604                                     HEX     E,9     ;Z
1919 003605                                     HEX     4,A     ;ASCII   BECOMES EBCDIC CENT SIGN
1920 003606                                     HEX     E,0     ;
1921 003607                                     HEX     5,F     ;ASCII   BECOMES EBCDIC "NOT" OPERATOR
1922 003610                                     HEX     4,F     ;ASCII   BECOMES EBCDIC VERTICAL BAR
1923 003611                                     HEX     6,D     ;UNDERLINE
1924                                    ;
1925                                    ;
1926                                    ; LOWER CASE
1927                                    ;
1928                                    ; 140
1929 003612                                     HEX     7,9     ;ACCENT
1930 003613                                     HEX     8,1     ;A
1931 003614                                     HEX     8,2     ;B
1932 003615                                     HEX     8,3     ;C
1933 003616                                     HEX     8,4     ;D
1934 003617                                     HEX     8,5     ;E
1935 003620                                     HEX     8,6     ;F
1936 003621                                     HEX     8,7     ;G
1937                                    ;
1938                                    ; 150
1939 003622                                     HEX     8,8     ;H
1940 003623                                     HEX     8,9     ;I
1941 003624                                     HEX     9,1     ;J
1942 003625                                     HEX     9,2     ;K
1943 003626                                     HEX     9,3     ;L
1944 003627                                     HEX     9,4     ;M
1945 003630                                     HEX     9,5     ;N
1946 003631                                     HEX     9,6     ;O
1947                                    ;
1948                                    ; 160
1949 003632                                     HEX     9,7     ;P
1950 003633                                     HEX     9,8     ;Q
1951 003634                                     HEX     9,9     ;R
```

```
1952 003635                                     HEX     A,2     ;S
1953 003636                                     HEX     A,3     ;T
1954 003637                                     HEX     A,4     ;U
1955 003640                                     HEX     A,5     ;V
1956 003641                                     HEX     A,6     ;W
1957                                    ;
1958                                    ; 170
1959 003642                                     HEX     A,7     ;X
1960 003643                                     HEX     A,8     ;Y
1961 003644                                     HEX     A,9     ;Z
1962 003645                                     HEX     C,0     ;LEFT BRACE
1963 003646                                     HEX     6,A     ;OR
1964 003647                                     HEX     D,0     ;RIGHT BRACE
1965 003650                                     HEX     A,1     ;TILDE
1966 003651                                     HEX     0,7     ;DEL
1967
1968
```

```
1970                                            .EVEN
1971                                            .ENDC
1972                                    ; ACTION INDEX TABLE - INDEX WITH ASCII CHAR < 40 OCTAL
1973                                    ; CONTENT IS ACTION CODE FOR THAT CHAR.  CODE IS ALWAYS
1974                                    ; EVEN FOR USE AS JUMP INDEX
1975
1976
1977 003652  000  002  002  ACTTAB: .BYTE 00,02,02,06,20,16,00,00   ;ASCII 00-07
     003655  006  020  016
     003660  000  000
1978 003662  000  000  000          .BYTE 00,00,00,00,00,00,00,00   ;ASCII 08-0F
     003665  000  000  000
     003670  000  000
1979 003672  012  000  000          .BYTE 12,00,00,00,00,22,14,04   ;ASCII 10-17
     003675  000  000  022
     003700  014  004
1980                                            .IF NDF R3780
1981 003702  000  000  000          .BYTE 00,00,00,00,00,00,00,10   ;ASCII 18-1F
     003705  000  000  000
     003710  000  010
1982                                            .ENDC
1983                                            .IF DF  R3780
1984                                            .BYTE 00,00,00,00,00,00,22,10   ;ASCII 18-1F
1985                                            .ENDC
1986
1987                                            .EVEN
1988                                    ;JUMP TABLE FOR SPECIAL ACTION CHARACTERS - RECEIVE MODE
1989 003712  004232'                    GTSAT:  GTBCC                   ;CHARACTER WAS NOT SPECIAL
1990 003714  005710'                            GTSTX                   ;STX/SOH
1991                                            .IF DF  R3780
1992                                            GTBCC           ;IRS-TREAT AS DATA
1993                                            .ENDC
1994 003716  005736'                            GTETB                   ;ETB
1995 003720  006002'                            GTETX                   ;ETX
1996 003722  006012'                            GTITB                   ;ITB
1997 003724  004232'                            GTBCC                   ;DLE
1998 003726  006070'                            GTSYN                   ;SYN
1999 003730  006040'                            GTENQ                   ;ENQ
2000 003732  005710'                            GTSTX                   ;EOT
2001 003734  004232'                            GTBCC                   ;NAK
2002                                    ;
2003                                    ;JUMP TABLE FOR SPECIAL ACTION CHARACTERS - 1ST CHARACTER RECEIVE MODE
2004 003736  005206'                    FRSAT:  GTFRT1                  ;CHARACTER WAS NOT SPECIAL
2005 003740  005274'                            SETMIP                  ;STX/SOH
2006 003742  005206'                            GTFRT1                  ;ETB
2007 003744  005206'                            GTFRT1                  ;ETX
2008 003746  005206'                            GTFRT1                  ;ITB
2009 003750  005414'                            SETDLE                  ;DLE
2010 003752  005410'                            GETCH1                  ;SYN
2011 003754  005652'                            SETENQ                  ;ENQ
2012 003756  005424'                            SETEOT                  ;EOT
2013 003760  005446'                            SETNAK                  ;NAK
```

```
                                        ;JUMP TABLE FOR SPECIAL ACTION CHARACTERS - TRANSMIT MODE
004022                          PTSAT:  PUTDS                   ;CHARACTER WAS NOT SPECIAL
004024                                  PTSTX                   ;STX/SOH
004026                                  PUTDS                   ;ETB
004030                                  PUTDS                   ;ETX
004032                                  PTITB                   ;ITB
004034                                  PTDLE                   ;DLE
004036                                  PTSYN                   ;SYN
004040                                  PTENQ                   ;ENQ
004042                                  PTEOT                   ;EOT
004044                                  PUTDS                   ;NAK
                                        .IF DF  R3780
                                        PTIRS                   ;IRS
                                        .ENDC

                                        .IF NZ  CA.EBC
                                ;EBCDIC-ASCII RECEIVE TRANSLATOR
004046  042704  177400          EBCRTR: BIC     #177400,CBBX
004052  010405                          MOV     CBBX,CBAX               ;SAVE ORIG. CHAR
004054  116404  003052'                  MOVB    ETA(CBBX),CBBX          ;GET ASCII EQUIVALENT
004060  000207                          RTS     PC
                                        .ENDC

                                ; CIRCULAR RECEIVE BUFFER (CRB) SERVICE ROUTINE
                                ; ENTERED FROM SCIF SYSTEM TIMER
                                ; FIRST, SAVE THE CURRENT STATE OF THE KG11

004062                          CRBSRV: .IF NDF RSTS
                                        .IF DF LSI11
                                        MOV     CRCACC,-(SP)            ;SAVE PARTIAL BCC
                                        MOV     CBC(R2),CRCACC          ;SET CURRENT PARTIAL BCC
                                        .IFF
004062  013746  170700                  MOV @#KGSTAT,-(SP)              ;STATUS
004066  013746  170702                  MOV @#KGBCCR,-(SP)              ;PARTIAL BCC
004072  012737  000133  170700          MOV #133,@#KGSTAT               ;CLEAR KG11 AND SET FOR LRC16
004100  016237  000122  170704          MOV CBC(R2),@#KGDATA            ;SET CURRENT PARTIAL BCC
004106  016201  000132                  MOV CCT(R2),R1                  ;SET CURRENT CODE TABLE POINTER
004112  016137  000034  170700          MOV KGC(R1),@#KGSTAT                    ;AND INIT KG11 FOR CURRENT CODE
                                        .ENDC
                                        .ENDC
                                        .IF DF RSTS!LSI11
                                        MOV     CCT(R2),R1              ;SET CURRENT CODE TABLE POINTER
                                        .ENDC
004120  005062  000062                  CLR IRC(R2)                     ;CLEAR THE INTERIM RECEIVE COUNT
                                ;NOW WE'RE READY TO SERVICE THE CRB
                                ; GET A CHAR FROM THE CRB AND PROCESS IT
004124                          GETCHR:
                                        .IF NZ  CRBINT
                                        MOV     @RBU,(SP)       ;GET UNLOADER ADDRESS
                                        JSR     PC,(SP)+        ;AND CALL IT
                                        .ENDC
                                        .IF Z   CRBINT

004124  004772  000004                  JSR PC,@RBU(R2)         ;ASK FOR ENTRY
                                        .ENDC
004130  100002                          BPL     GETCHA          ;ANY ENTRIES?
004132  000167  000640                  JMP     GTNONE          ;NO
```

```
2072 004076  102003                    GETCHA: BVC GET1                 ;CRB OVERRUN INDICATED?
2073 004100  052762  000040  000012            BIS #COVR,RSW(R2)         ;SET STATUS FLAG IF SO
2074 004106  032704  020000            GET1:   BIT #20000,CBBX           ;TEST FOR RCV OVERRUN
2075 004112  001405                            BEQ GET1A                 ;SET STATUS FLAG IF OVERRUN
2076 004114  042704  020000                    BIC #20000,CBBX           ;CLEAR OVERRUN FLAG IN ENTRY
2077 004120  052762  000100  000012            BIS #ROVR,RSW(R2)
2078 004126  032762  000200  000012  GET1A:    BIT     #SRTOF,RSW(R2)    ;SLAVE RCVE TIMEOUT?
2079 004134  001405                            BEQ     GET1B             ;NO
2080 004136  042762  000200  000012            BIC     #SRTOF,RSW(R2)
2081 004144  000167  001700                    JMP     GTEOT             ;GENERATE AB END RCVE UIB
2082 004150  032762  000400  000012  GET1B:    BIT     #FBF,RSW(R2)      ;FLUSHING BUFFER?
2083 004156  001342                            BNE     GETCHR            ;YES - IGNORE
2084 004160  005262  000062                    INC IRC(R2)               ;INDICATE CHAR CAME IN THIS TIME
2085 004164  105762  000127                    TSTB BCL(R2)              ;ARE WE IN BCC PHASE?
2086 004170  001402                            BEQ GET2
2087 004172  000167  001710                    JMP BCCHK                 ;YES
2088 004176  004771  000040            GET2:   JSR PC,@ATR(R1)           ;NO - GO TO ACTIVE TRANSLATE ROUTINE
2089 004202  032762  100000  000010            BIT #MIP,LSW(R2)          ;
2090 004210  001002                            BNE GET5                  ;NOT THE FIRST CHAR - PROCEED
2091 004212  000167  000752                    JMP GTFRST                ;FIRST CHAR - GO CHECK IT
2092 004216  120427  000037            GET5:   CMPB CBBX,#37             ;IS IT IN RANGE OF SPECIAL CHARS?
2093 004222  003440                            BLE GET3                  ;YES - GO CHECK FURTHER
2094 004224  042762  000200  000006            BIC #DLEF,LCW(R2)
2095 004232  032762  000200  000010  GTBCC:    BIT #BCCA,LSW(R2)                 ;INCLUDE IT IN BCC?
2096 004240  001402                            BEQ GET4                  ;NO
2097                                           .IF DF LSI11
2098                                           MOVB    CBAX,CRCCHR               ;RUN IT THROUGH BCC CALC
2099                                           JSR     PC,$CRC16
2100                                           .IFF
2101 004242  110537  170704                    MOVB CBAX,@#KGDATA        ;RUN IT THROUGH KG11
2102                                           .ENDC
2103 004246  032762  000020  000012  GET4:    BIT #UOVR,RSW(R2)                 ;USER BUFFER OVERFLOW?
2104 004254  001303                            BNE GETCHR                ;YES.  IGNORE FUTHER CHARS
2105 004256  005762  000024                    TST ABC(R2)               ;ROOM FOR THIS CHAR?
2106 004262  001444                            BEQ GTUOF                         ;NO.  GO SET OVERFLOW FLAG
2107 004264  032762  040000  000010            BIT #ANF,LSW(R2)                  ;IF ACKNOWLEDGE, DON'T BUFFER IT
2108 004272  001274                            BNE GETCHR
2109 004274  032762  001000  000006            BIT #NAT.LCW(R2)                  ;IF NATIVE MODE....
2110                                           .IF NDF RSTS
2111 004302  001401                            BEQ GET9
2112 004304  010504                            MOV CBAX,CBBX                     ;... BUFFER ORIGINAL CHAR
2113 004306                            GET9:
2114 004306  110472  000022                    MOVB    CBBX,@ARA(R2)             ;PUT CHAR IN USER BUFFER
2115                                           .ENDC
2116                                            IF DF RSTS
2117                                           BNE     5$                ;YES
2118                                           .GLOBL  EXTFLG
2119                                           TSTB    EXTFLG            ;CHECK FOR RSTS-TO-RSTS MODE
2120                                                                     ;PUT IN BECAUSE OF RACE CONDITION
2121                                                                     ;EDIT 31
2122                                           BPL     GET9              ;NOT SET
2123                                   5$:     MOV     CBAX,CBBX         ;BUFFER ORIGINAL CHARACTER
2124                                   GET9:
2125                                           MOV     SAVR6,KISAR6              ;RESTORE APR 6 JUST IN CASE
2126                                           MOVB    CBBX,@SAVADD              ;PUT CHAR INTO POTENTIALLY
2127                                                                             ;MAPPED BUFFER AREA
2128                                           INC     SAVADD                    ;BUMP POINTER
```

```
2129
2130 004312  005262  000022                .ENDC
2131 004316  005362  000024                INC ARA(R2)                ;UPDATE POINTER AND COUNT
2132 004322  000660                        DEC ABC(R2)
                                           BR GETCHR
```

```
2134                                    ; PROCESS A POSSIBLE SPECIAL CHAR
2135 004324  116400  003652'    GET3:   MOVB ACTTAB(CBBX),R0    ;FETCH THE ACTION CODE
2136 004330  032762  100000  000006      BIT #RCVT,LCW(R2)               ;TRANSPARENCY
2137 004336  001414                      BEQ GTNSAR
2138 004340  032762  000200  000006      BIT #DLEF,LCW(R2)
2139 004346  001005                      BNE GNX1                ;IF DLEF SET, PROCESS THE CHAR
2140 004350  120427  000020              CMPB CBBX,#DLE          ; IF NOT, LOOK FOR ONLY DLE
2141 004354  001326                      BNE GTBCC               ;OTHERS TREATED AS DATA
2142 004356  000167  001032              JMP SETDLE              ;DLE SETS THE FLAG
2143
2144 004362  042762  000200  000006  GNX1:   BIC #DLEF,LCW(R2)
2145 004370  000170  003712'    GTNSAR: JMP @GTSAT(R0)          ;GO TO INDIVIDUAL ACTION ROUTINE
2146
2147
2148                                    ;USER BUFFER HAS OVERFLOWED
2149 004374  032762  000001  000010  GTUOF:  BIT #RBA,LSW(R2)                ;.. IF BECAUSE THERE WASN'T ONE
2150 004402  001003                      BNE GOUOF                       ;NO - PROCEED
2151 004404  052762  000002  000012      BIS #NUB,RSW(R2)                ;SET 'NO USER BUFFER' FLAG
2152 004412  052762  000020  000012  GOUOF:  BIS #UOVR,RSW(R2)           ;SET FLAG
2153 004420  000621                      BR GETCHR               ;AND CONTINUE DISCARDING CHARS
2154
2155                                    ;ROUTINE TO DEQUEUE SLAVE STATION RECEIVE TIMER
2156 004422  105062  000153      CSSRT:  CLRB    SRC(R2)         ;RESET COUNTER TO MINUS ONE
2157 004426  042762  000400  000010      BIC     #SRTA,LSW(R2)   ;CLEAR SLAVE RECV. TIMER ACTIVE FLAG
2158 004434  000207                      RTS     PC              ;RETURN
2159
2160
```



```
2161                                    ;ROUTINE TO ENQUEUE SLAVE STATION RECEIVE TIMER
2162 004436  012762  177777  000152      MOV     #-1,SRC(R2)     ;SET COUNTER TO MINUS ONE
2163 004444  052762  000400  000010      BIS     #SRTA,LSW(R2)   ;SET SLAVE RECV. TIMER ACTIVE FLAG
2164 004452  000207                      RTS     PC              ;RETURN
2165
2166                                    ;ROUTINE TO DEQUEUE SLAVE STATION TRANSMIT TIMER
2167 004454  105062  000151      CSSTT:  CLRB    SRC+1(R2)       ;RESET COUNTER TO MINUS ONE
2168 004460  042762  000100  000010      BIC     #STTA,LSW(R2)   ;CLEAR SLAVE XMIT. TIMER ACTIVE FLAG
2169 004466  000207                      RTS     PC              ;RETURN
2170
2171                                    ;ROUTINE TO ENQUEUE SLAVE STATION TRANSMIT TIMER
2172 004470  012762  177777  000152      MOV     #-1,SRC(R2)     ;SET COUNTER TO MINUS ONE
2173 004476  052762  000100  000010      BIS     #STTA,LSW(R2)   ;SET SLAVE XMIT. TIMER ACTIVE FLAG
2174 004504  000207                      RTS     PC              ;RETURN
2175
2176                                    ;ROUTINE TO DEQUEUE MASTER STATION TIMER
2177 004506  105062  000151      CMSTT:  CLRB    SRC+1(R2)       ;RESET COUNTER TO MINUS ONE
2178 004512  042762  000040  000010      BIC     #MTTA,LSW(R2)   ;CLEAR MASTER TIMER ACTIVE FLAG
2179 004520  000207                      RTS     PC              ;RETURN
2180
2181                                    ;ROUTINE TO ENQUEUE MASTER STATION TIMER
2182 004522  012762  177777  000152      MOV     #-1,SRC(R2)     ;SET COUNTER TO MINUS ONE
2183 004530  052762  000040  000010      BIS     #MTTA,LSW(R2)   ;SET MASTER TIMER ACTIVE FLAG
2184 004536  000207                      RTS     PC              ;RETURN
2185
2186                                    ;ROUTINE TO DEQUEUE MASTER STATION POLL TIMER
2187 004540  105062  000150      CMSPT:  CLRB    SRC(R2)         ;RESET COUNTER TO MINUS ONE
2188 004544  042762  000020  000010      BIC     #MPTA,LSW(R2)   ;CLEAR MASTER POLL TIMER ACTIVE FLAG
2189 004552  000207                      RTS     PC              ;RETURN
2190
2191                                    ;ROUTINE TO ENQUEUE MASTER STATION POLL TIMER
2192 004554  012762  177777  000152      MOV     #-1,SRC(R2)     ;SET COUNTER TO MINUS ONE
2193 004562  052762  000020  000010      BIS     #MPTA,LSW(R2)   ;SET MASTER POLL TIMER ACTIVE FLAG
2194 004570  000207                      RTS     PC              ;RETURN
2195
2196
```

```
                                        ;ROUTINE TO BUILD TRANSMIT MESSAGE POINTERS FOR ENQ

                                        ;PROVIDES FOR MASTER STATION RECEIVE TIMEOUT, RECEIVED WACK,
                                        ;AND GARBLED MESSAGE RESPONSE

                                        ;COME HERE IF TIMEOUT (ENTERED FROM SCIP TIMER)
                                        ;SCIP HAS SAVED REGISTERS

004436  052762  000100  000014  MSRT0:  BIS     #MTOUT,TSW(R2)  ;SET FLAG THAT WE HAD A TIMEOUT
004444  032762  020000  000010          BIT     #AEF,LSW(R2)    ;STILL WAITING FOR ACK
004452  001426                          BEQ     MSRTOX          ;NO - IGNORE TIMEOUT

004454  032762  000400  000006          BIT     #INI,LCW(R2)    ;STILL IN MESSAGE TRANSFER
004462  001422                          BEQ     MSRTOX          ;NO - IGNORE TIMEOUT
004464                          MSRT1:  .IF DF RSTS
                                        .GLOBL  CLOSD
                                        MOV     CLOSD,R5        ;PUT DDB ADD IN R5 *******EDIT 32
                                        INCB    12(R5)          ;INCRMNT # OF TRANSMIT TIMEOUTS
                                        .IFF
                                        .GLOBL  XMTTMT                  ;****EDIT 40
004464  005267  000006                  INC     XMTTMT          ;INCREMENT # OF TRANSMIT TIMEOUTS
                                        .ENDC
004470  105362  000140                  DECB    EKT(R2)         ;ENQ COUNT EXPIRED?

004474  100007                          BPL     MSRTOA          ;NO - SEND ENQ

                                        ;ENQ RETRY COUNT REACHED - PERFORM ABNORMAL
                                        ;TRANSMIT TERMINATION

004476  052762  100000  000014          BIS     #NAKF,TSW(R2)   ;ABNORMAL TRANSMIT TERMINATION
                                        .IF NDF RSTS
004504  005000                          CLR     R0              ;PUT LINE # IN R0
004506  004767  000532'                  JSR     PC,TDENT        ;STORE TRANSMIT DONE UIB ENTRY
004512  000207                          RTS     PC              ;RETURN FROM TIMER
                                        .IFF
                                        JMP     TDENTO
                                        .ENDC
```

```
2201                                    ;TRANSMIT ENQ TO REQUEST RESPONSE
2202
2203 004514  016201  000132            MSRTOA: MOV     CCT(R2),R1      ;GET CURRENT CODE TABLE POINTER
2204 004520  004767  000006                    JSR     PC,BENQ         ;BUILD ENQ MESSAGE POINTERS
2205 004524  004767  000150'                   JSR     PC,ACKOUT       ;TRANSMIT ENQ WITHOUT DELAY
2206
2207 004530  000207                    MSRTOX: RTS     PC              ;RETURN TO SCIP TIMER
2208
2209                                    ;ROUTINE TO BUILD MESSAGE POINTERS FOR ENQ
2210
2211 004532  016262  000100  000144    BENQ:   MOV     T2A(R2),TT2A(R2)        ;MUST SAVE CURRENT MESSAGE
2212 004540  016262  000102  000146            MOV     T2C(R2),TT2C(R2)        ;POINTERS IN CASE
2213 004546  016262  000104  000150            MOV     T3A(R2),TT3A(R2)        ;RETRANSMISSION IS REQUIRED
2214
2215                                                                    ;VALUES WILL BE RESTORED WHEN
2216                                                                    ;THE ISR OUTPUT STARTUP
2217                                                                    ;CALL IS COMPLETED
2218
2219 004554  052762  001000  000010            BIS     #RDBF,LSW(R2)           ;'RESTORE BUFFER POINTERS' FLAG
2220
2221 004562  010162  000100                    MOV     R1,T2A(R2)
2222 004566  062762  000022  000100            ADD     #CENQ,T2A(R2)           ;POINTER TO 'ENQ' 'PAD'
2223 004574  012762  000002  000102            MOV     #2,T2C(R2)              ;SET UP COUNT
2224 004602  005062  000104                    CLR     T3A(R2)                 ;TERMINATE ADDRESS/COUNT CHAIN
2225
2226 004606  000207                            RTS     PC                      ;ENQ READY TO GO
2227
2228                                    ;SUBROUTINE TO START THE 1/2 SECOND INTERVAL TIMER
2229 004610                             INTVTM:
2230 004610  005000                            CLR     R0              ;GET LINE NUMBER AS BYTE INDEX
2231 004612  010546                            MOV     R5,-(SP)        ;SAVE RELEVANT REGISTERS
2232 004614  010446                            MOV     R4,-(SP)        ; ...
2233                                           .IF DF  UAPTIM
2234 004616  016704  173164                    MOV     TICINT,R4       ;GET  # OF TICS IN AN INTERVAL
2235                                           .IFF
2236                                           MOV     #TICINT,R4      ;GET THE # OF TICS IN AN INTERVAL
2237                                           .ENDC
2238 004622  012705  004656'                   MOV     #INTTIC,R5      ;TIMEOUT ROUTINE ADDRESS
2239 004626  004767  000000G                   JSR     PC,CTIME        ;START THE TIMER
2240 004632  012604                    INTVT1: MOV     (SP)+,R4        ;RESTORE REGISTERS
2241 004634  012605                            MOV     (SP)+,R5        ;..
2242 004636  000207                            RTS     PC              ; RETURN
2243                                    ;SUBROUTINE TO START SLAVE STATION RECEIVE TIMER
2244
2245 004640                             SSRTM:
2246                                           .IF DF  UAPTIM
2247 004640  116762  173140  000153            MOVB    SSRT,SRC(R2)            ;SET TIMER INTERVAL
2248                                           .IFF
2249                                           MOVB    #SSRT,SRC(R2)           ;SET TIMER INTERVAL
2250                                           .ENDC
2251 004646  052762  000400  000010            BIS     #SRTA,LSW(R2)           ;SET SLAVE TIMER ACTIVE FLAG
2252 004654  000207                            RTS     PC                      ;RETURN
2253
2254                                    ;ROUTINE TO SERVICE INTERVAL TIMER
2255 004656  105762  000153            INTTIC: TSTB    SRC(R2)                 ;SLAVE RECEIVE TIMER GOING?
2256 004662  001405                            BEQ     INTC1                   ;NO
2257 004664  105362  000153                    DECB    SRC(R2)                 ;COUNT DOWN ONE INTERVAL
```

```
2258 004670  001002                        BNE     INTC1                   ;HASN'T EXPIRED YET
2259 004672  004767  000022                JSR     PC,SSRTO                ;EXPIRED, CALL TIMEOUT ROUTINE
2260 004676  105762  000115        INTC1:  TSTB    MRC(R2)                 ;MASTER RECEIVE TIMER GOING?
2261 004702  001405                        BEQ     INTC2                   ;NO
2262 004704  105362  000115                DECB    MRC(R2)                 ;COUNT DOWN ONE INTERVAL
2263 004710  001002                        BNE     INTC2                   ;HASN'T EXPIRED  YET
2264 004712  004767  177520                JSR     PC,MSRTO                ;EXPIRED, CALL TIMEOUT ROUTINE
2265 004716                        INTC2:
2266 004716  000734                        BR      INTVTM
2267
2268
2269                                ;SLAVE STATION RECEIVE TIMEOUT - (FROM SCIP SYSTEM TIMER)
2270
2271 004720  042762  000400  000010  SSRTO:  BIC     #SRTA,LSW(R2)           ;CLEAR SLAVE TIMER ACTIVE FLAG
2272 004726  052762  000200  000012          BIS     #SRTOF,RSW(R2)          ;SET SLAVE RECEIVE TIME OUT FLAG
2273                                                                        ;RECEIVE WILL BE ABORTED AND NAK
2274                                                                        ;SENT ON NEXT CRB SRV PASS
2275 004734  000207                        RTS     PC                      ;RETURN TO SCIP TIMER
```

```
2277 004736                                GTNONE: .IF NDF RSTS
2278 004736  005762  000062                       TST IRC(R2)              ;DID WE GET ANY THIS TIME?
2279 004742  001406                               BEQ     GTNO2
2280 004744  105762  000127                GTNO1: TSTB BCL(R2)                     ;IF NOT IN BCC PHASE...
2281 004750  001003                               BNE GTNO2
2282                                               .IF DF LSI11
2283                                               MOV     CRCACC,CBC(R2)          ;SAVE THE PARTIAL BCC
2284                                               .IFF
2285 004752  013762  170702  000122                MOV @#KGBCCR,CBC(R2)            ;..SAVE THE PARTIAL BCC
2286                                               .ENDC
2287                                               .ENDC
2288                                       ; NOW SET UP ANOTHER CRB SERVICE TIMER REQUEST FOR THIS LINE
2289 004760                               GTNO2:
2290                                               .IF Z   CRBINT
2291 004760  032762  000002  000010                BIT #CSA,LSW(R2)         ;IF TIMER NO LONGER ACTIVE...
2292 004766  001411                               BEQ GTNO3                ;..  DON'T REQUEST ANOTHER
2293                                               .ENDC
2294 004770  005000                               CLR     R0               ;RECREATE LINE #
2295 004772  116204  000120                       MOVB TCT(R2),R4          ;SET TIMER COUNT
2296 004776  012705  004022'                      MOV #CRBSRV,R5           ;SET TIMER ROUTINE POINTER
2297 005002  004767  0000006                      JSR PC,CTIME             ;REQUEST TIMER ENTRY
2298                                               .IF NDF RSTS
2299 005006  100001                               BPL     GTNO3            ;TIMER QUEUE FULL?
2300 005010  000240                               THALT                    ;YES-SYSTEM ERROR
2301                                       ; RESTORE THE KG11 AND EXIT
2302 005012                               GTNO3:  .IF DF LSI11
2303                                               MOV     (SP)+,CRCACC             ;RESTORE PARTIAL BCC
2304                                               .IFF
2305 005012  012737  000133  170700                MOV #133,@#KGSTAT                ;CLEAR IT AND SELECT LRC16
2306 005020  012637  170704                       MOV (SP)+,@#KGDATA               ;RESTORE ORIGINAL BCC
2307 005024  012637  170700                       MOV (SP)+,@#KGSTAT               ;AND STATUS CONTENTS
2308                                               .ENDC
2309                                               .ENDC
2310                                               .IF DF RSTS
2311                                       ;EXIT
2312                                       GTNO3:
2313                                               .ENDC
2314 005030  042762  000400  000012                BIC #FBF,RSW(R2)                 ;BUFFER IS NOW FLUSHED
2315 005036  000207                               RTS PC                   ;RETURN TO TIMER
2316
2317
2318
2319 005040  004767  173060                ARFEND: JSR PC,C.BRCD            ;GENERATE 'RECEIVE DONE' UIB ENTRY
2320 005044  032762  000400  000010                BIT     #SRTA,LSW(R2)            ;RCVE TIMER ACTIVE?
2321 005052  001402                               BEQ     ARF1                     ;NO
2322 005054  004767  177342                       JSR     PC,CSSRT                 ;CLEAR TIMER
2323 005060  016201  000132                ARF1:  MOV CCT(R2),R1                   ;RESTORE CCT POINTER
2324 005064  042762  112200  000010                BIC #MIP!BCCA!ITBA!XAC,LSW(R2)   ;CLEAR PROCESS FLAGS
2325 005072  042762  000200  000006                BIC #DLEF,LCW(R2)
2326 005100  042762  000100  000014                BIC     #MTOUT,TSW(R2)           ;CLEAR MASTER TIMEOUT FLAG
2327                                               .IF Z   CRBINT
2328 005106  052762  000400  000012                BIS #FBF,RSW(R2)                 ;FLUSH BUFFER OF TRAILING CHARS
2329 005114  000167  176744                       JMP GETCHR
2330                                               .ENDC
2331                                               .IF NZ  CRBINT
2332                                       ARF2:   MOV     RBU(R2),-(SP)
2333                                               ADD     #2,@SP
```

```
2334                                            JSR     PC,@(SP)+
2335                                            BR      GTNONE
2336                                            .ENDC
2337
2338                                    ;WE RAN OUT OF CHARS WITHOUT RECOGNIZABLE END
2339 005120  032762  020000  000010    GTNOPE: BIT #AEF,LSW(R2)        ;WERE WE EXPECTING ACKNOWLEDGE
2340 005126  001412                            BEQ GENQ10                      ;NO. GENERATE 'AB END' RCVE UIB
2341 005130  105062  000115                    CLRB    MRC(R2)         ;RESET MASTER RECEIVE COUNTER
2342                                            .IF DF RSTS
2343                                            MOV     CLOSD,R5        ;PUT ADDR OF DDB IN R5 ****EDIT 34
2344                                            INCB    12(R5)          ;INCRMT TIMEOUT NAK COUNTER
2345                                            .ENDC
2346 005134  105362  000140                    DECB EKT(R2)    ;OK TO SEND ENQ?
2347 005140  100556                            BMI TRANER                      ;NO - WE'VE DONE 3
2348 005142  004767  177364                    JSR     PC,BENQ         ;BUILD ENQ MSG POINTER
2349 005146  004767  000150'          REPEAT: JSR PC,ACKOUT                   ;SEND OUT THE ENQ
2350 005152  000742                            BR ARF1                         ;GO FLUSH TRAILING CHARACTERS
2351
2352 005154  000167  000670          GENQ10: JMP GTEOT
2353                                    ;
2354                                    ; WACK RECEIVED AS MESSAGE RESPONSE
2355                                    ;
2356 005160                            SETWAK: .IF DF RSTS
2357                                            BISB    #1,EXTFLG       ;SET FLAG SO RSTS KNOWS IS BEING WAKED
2358                                                                    ;EDIT 30
2359                                            .ENDC
2360 005160  012762  000011  000140            MOV     #ENQCNT+1,EKT(R2)       ;RESET ENQ COUNTER
2361 005166  000734                            BR      ARF1            ;LET TIMEOUT CAUSE ENQ TO BE SENT
```

```
                                        ; PROCESS THE FIRST CHAR OF MESSAGE

005170  120427  000037          GTFRST: CMPB    CBBX,#37        ;IS IT IN RANGE OF SPECIAL CHARACTERS?
005174  003004                          BGT     GTFRT1          ;NO
005176  116400  003652'                MOVB    ACTTAB(CBBX),R0 ;FETCH THE ACTION CODE
005202  000170  003736'                JMP     @FRSAT(R0)      ;AND GO TO SPECIAL ACTION ROUTINE
005206  120427  000177          GTFRT1: CMPB CBBX,#PAD          ;IF PAD, IGNORE IT
005212  001476                          BEQ GETCH1
                                ;NOTE - THESE NEXT TESTS USE ORIG. CHAR - STILL IN CBAX
005214  032762  000200  000006          BIT #DLEF,LCW(R2)       ;IF DLEF NOT SET, DONE TESTING
005222  001417                          BEQ SETARF              ;DIDN'T RECOGNIZE THE CHAR
                                        .IF DF RSTS
                                        BICB    #1,EXTFLG       ;MAKE SURE FLAG USUALLY CLEAR
                                        .ENDC
005224  042762  000200  000006          BIC #DLEF,LCW(R2)
005232  120562  000134                  CMPB CBAX,CAK(R2)       ;ACK?
005236  001532                          BEQ SETACK              ;YES
005240  120562  000135                  CMPB    CBAX,CAK+1(R2)  ;WRONG ACK?
005244  001566                          BEQ     SETWRG          ;YES
005246  120561  000001                  CMPB CBAX,CWAK(R1)              ;NO - WACK?
005252  001742                          BEQ SETWAK              ;YES
005254  120561  000000                  CMPB CBAX,CRVI(R1)              ;NO - RVI
005260  001531                          BEQ SETRVI              ;YES

                                ;HERE IF THIS IS A LEADING GRAPHIC

005262  105362  000152          SETARF: DECB    LGC(R2)         ;DECREMENT COUNTER
005266  100564                          BMI     STARF1          ;ONE TOO MANY
005270  000167  176570                  JMP     GETCHR          ;OTHERWISE IGNORE IT AND KEEP READING

005274  052762  100200  000010  SETMIP: BIS #MIP!BCCA,LSW(R2)  ;SET MIP AND ALLOW BCC ACCUMULATION
005302  004767  177332                  JSR     PC,SSRTM        ;START SLAVE RCVE TIMER
005306  032762  000200  000006          BIT #DLEF,LCW(R2)       ;PRECEDED BY DLE?
005314  001411                          BEQ INTBCC              ;NO.  INITIALIZE BCC
005316  052762  100000  000006          BIS #RCVT,LCW(R2)       ;YES.  SET UP FOR TRANSPARENCY
005324  052762  001000  000012          BIS #TRNREC,RSW(R2)     ;INDICATE TO USER THAT THIS IS A
                                                                ;  TRANSPARENT RECORD ***EDIT 21***
005332  042762  000200  000006          BIC #DLEF,LCW(R2)
005340  005062  000122          INTBCC: CLR CBC(R2)             ;COMPUTED BCC
005344  005062  000124                  CLR RBC(R2)             ;RECIEVED BCC
                                        .IF DF RSTS
                                        MOV #121,@#KGSTAT       ;INITIALIZE KG BCC REGISTER
                                        .IFF
                                        .IF DF LSI11
                                        CLR     CRCACC
                                        .IFF
005350  052737  000020  170700          BIS #20,@#KGSTAT        ;CLEAR KG BCC REGISTER
                                        .ENDC
                                        .ENDC
005356  032762  002000  000010          BIT #ITBA,LSW(R2)       ;ITB ACTIVE?
005364  001411                          BEQ GETCH1              ;NO - NO MORE TO DO HERE
005366  032762  100000  000006          BIT #RCVT,LCW(R2)       ;YES - RUN STX THRU KG11 THIS TIME
005374  001403                          BEQ SETM1
                                        .IF DF LSI11
                                        MOVB    CDLE(R1),CRCCHR ;ALSO A DLE IF IN TRANSPARENCY
                                        JSR     PC,$CRC16
                                        .IFF
```

```
2420 005376  116137  000010  170704          MOVB CDLE(R1),@#KGDATA  ;ALSO A DLE IF IN TRANSPARENCY
2421                                         .ENDC
2422                                         .IF DF RSTS
2423                                 1$:     TSTB    @#KGSTAT        ;DONE YET?
2424                                         BPL     1$              ;NOPE, WILL BE DONE SOON...
2425                                         .ENDC
2426 005404                         SETM1:  .IF DF LSI11
2427                                         MOV     CBAX,CRCCHR     ;THIS IS THE STX
2428                                         JSR     PC,$CRC16       ;ACCUMLATE IN BCC
2429                                         .IFF
2430 005404  110537  170704                 MOVB CBAX,@#KGDATA      ;THIS IS THE STX
2431                                         .ENDC
2432 005410  000167  176450          GETCH1: JMP GETCHR              ;GO GET NEXT CHAR
2433
2434 005414  052762  000200  000006  SETDLE: BIS #DLEF,LCW(R2)       ;SET THE DLE FLAG
2435 005422  000772                         BR GETCH1
2436
2437 005424  032762  000200  000006  SETEOF: BIT #DLEF,LCW(R2)       ;DLE EOT?
2438 005432  001450                         BEQ SETEOF              ;NO - GO BACK TO CONTROL MODE
2439 005434  052762  104000  000012          BIS #ARF!DSC,RSW(R2)    ;YES SET DISCONNECT FLAG
2440 005442  000167  177372                 JMP ARFEND              ;AND CREATE UIB ENTRY
2441
2442 005446                          SETNAK: .IF DF RSTS!RSX
2443 005446  105062  000115                 CLRB    MRC(R2)         ;CLEAR MASTER RECEIVE TIMER
2444                                         .IFF
2445                                         JSR     PC,CMSRT        ;CLEAR MASTER STATION RCVE TIMER
2446                                         .ENDC
2447 005452  032762  004000  000010          BIT     #NEF,LSW(R2)    ;WERE WE EXPECTING A NAK? - EDIT 17
2448 005460  001006                         BNE     TRANER          ;YES - DON'T RETRANSMIT - EDIT 17
2449                                         .IF DF RSTS
2450                                         MOV     CLOSD,R5        ;SET R5 TO ADDR OF DDB *******EDIT 34
2451                                         INCB    10(R5)          ;COUNT TRANSMIT NAKS
2452                                         .ENDC
2453 005462  105362  000141                 DECB RKT(R2)
2454 005466  002403                         BLT TRANER
2455 005470  004767  000212'         SETN1:  JSR PC,ARRCV            ;RESET THE LINE
2456 005474  000624                         BR REPEAT
2457 005476  052762  100000  000014  TRANER: BIS #NAKF,TSW(R2)       ;SET THE NAK FLAG
2458 005504                          TRAN1:  .IF NDF RSTS
2459 005504  010300                         MOV R3,R0               ;CREATE LINE #
2460 005506  006200                         ASR R0
2461 005510  004767  000532'                JSR PC,TDENT            ;CREATE 'XMIT DONE' UIB ENTRY
2462                                         .IFF
2463                                         JSR PC,TDENTO           ;CREATE 'XMIT DONE' UIB ENTRY
2464                                         .ENDC
2465 005514  004767  000212'         TRAN2:  JSR     PC,ARRCV        ;RESET THE LINE
2466 005520  000167  177334                 JMP ARF1
2467
2468 005524  042762  177377  000014  SETACK: BIC     #177377,TSW(R2) ;SET VALID STATUS RETURN
2469 005532                          SETAC1:
2470 005532  105062  000115                 CLRB    MRC(R2)         ;CLEAR MASTER RECEIVE TIMER
2471 005536  000362  000134                 SWAB CAK(R2)            ;RECEIVED VALID ACK - TOGGLE TEST WORD
2472 005542  000760                         BR TRAN1                ;..AND GO GENERATE 'TRANSMIT DONE'
2473
2474 005544  052762  040000  000014  SETRVI: BIS #RVI,TSW(R2)       ;SET RVI FLAG IN STATUS
2475 005552  000767                         BR SETAC1
2476
```

```
2477 005554                                SETEOF:
2478 005554  032762  002000  000014                BIT     #BSNT,TSW(R2)           ;WAS EOT IN RESPONSE TO MASTER BID?
2479 005562  001402                                BEQ     5$                      ;NO ***EDIT 41
2480 005564  000167  176274                        JMP     GETCHR                  ;YES - IGNORE IT
2481 005570                                5$:
2482 005570  042762  001420  0000[illegible]       BIC #MSS!INI!NAT,LCW(R2)        ;GO BACK TO CONTROL MODE
2483 005576  052762  002000  000[illegible]2       BIS #C.BEOF,RSW(R2)     ;SET EOF FLAG
2484 005604  105062  000136                        CLRB    AKT(R2)         ;RESET SLAVE ACK COUNTER
2485 005610  016162  000052  000134                MOV     CAKO(R1),CAK(R2)        ;RESET MASTER ACK COUNTER
2486 005616  000167  177216                        JMP ARFEND              ;AND NOTIFY USER
2487                                       ;
2488                                       ;HERE IF WE RECEIVE THE OTHER ACK FROM WHAT WE EXPECTED
2489 005622  032762  000100  000014  SETWRG: BIT     #ATOUT,TSW(R2)  ;HAS THERE BEEN A MASTER TIMEOUT?
2490 005630  001306                        BNE     SETNAK                  ;YES, TREAT AS A NAK A RETRANSMIT
2491                                                                        ;LAST BLOCK
2492 005632  004767  176626                        JSR     PC,MSRT1                ;NO - REQUEST RESEND OF RESPONSE
2493 005636  000726                        BR      TRAN2                   ;GO USE TIMEOUT CODE TO ABORT TRANSMIT IF NEEDED
2494                                       ;
2495 005640                                STARF1:                         ;HERE IF TOO MANY LEADING GRAPHICS
2496                                                                       ;SEEN OR GARBLED ACK RECEIVED WITH
2497                                                                       ;NO TIMEOUT
2498 005640  112762  000007  000152                MOVB    #LOGRPH,LGC(R2) ;RESET LEADING GRAPHICS COUNTER
2499 005646  000167  000416                        JMP     ENQZ                    ;CLEAR RECEIVE ACTIVE AND GO INTO SEARCH SYNC
2500                                                                       ;MODE.  THEN FLUSH THE CRB
2501 005652  032762  000400  000006  SETENQ: BIT #INI,LCW(R2)                ;WHICH PROCEDURE
2502 005660  001553                        BEQ     INIENQ                  ;INIT
2503                                               .IF DF RSTS
2504                                               MOV     CLOSD,R5                ;SET R5 TO DDB ******EDIT 32
2505                                               INCB    13(R5)                  ;COUNT RECEIVED ENQ'S
2506                                               .IFF
2507                                               .GLOBL  RCVTMT                  ;EDIT 40******
2508 005662  005267  000006                        INC     RCVTMT                  ;COUNT RECEIVED ENQ'S
2509                                               .ENDC
2510 005666  032762  000004  000010                BIT     #AAF,LSW(R2)    ;ACK AVAILABLE TO ANSWER OTHER SIDE?
2511 005674  001404                        BEQ     SETJ                    ;NO, IGNORE THE ENQ
2512 005676  032762  000020  000006                BIT #MSS,LCW(R2)                ;MESSAGE PROC - WHICH MODE?
2513 005704  001671                        BEQ SETN1                       ;SLAVE - REPEAT LAST ACK/NAK
2514 005706  000570                SETJ:   BR      ENQZ                    ;MASTER - IGNORE, PENDING CONTENTION
```

```
2516                                            ;PROCESS STX OR SOH IN TEXT
2517 005710  032762  100000  000006  GTSTX:    BIT     #RCVT,LCW(R2)           ;STX IS ACCUMULATED AFTER 1ST RECORD
2518 005716  001403                             BEQ     STXBCC                  ;IF TRANSPARENT ...
2519                                            .IF DF LS111
2520                                            MOVB    CDLE(R1),CRCCHR         ;THROW IN THE DLE TOO
2521                                            JSR     PC,$CRC16               ;
2522                                            .IFF
2523 005720  116137  000010  170704             MOVB    CDLE(R1),@#KGDATA       ;THROW IN THE DLE TOO
2524                                            .ENDC
2525                                            .IF DF RSTS
2526                                    1$:     TSTB    @#KGSTAT                ;DONE YET?
2527                                            BPL     1$                      ;NO, BUT IT WILL SOON. .
2528                                            .ENDC
2529 005726                             STXBCC: .IF DF LS111
2530                                            MOVB    CBAX,CRCCHR     ;
2531                                            JSR     PC,$CRC16               ;ACCUMLATE IN BCC
2532                                            .IFF
2533 005726  110537  170704                     MOVB    CBAX,@#KGDATA
2534                                            .ENDC
2535 005732  000167  176126             GTGET:  JMP     GETCHR
2536
2537                                            ;PROCESS ETB
2538 005736  052762  020000  000012  GTETB:    BIS     #BEND,RSW(R2)           ;SET 'BLOCK END' FLAG
2539 005744  042762  002000  000010  GTETO:    BIC     #ITBA,LSW(R2)           ;CLEAR 'ITB ACTIVE' FLAG
2540 005752  042762  100000  000006  GTET1:    BIC     #RCVT,LCW(R2)           ;CLEAR PROCESS FLAGS
2541 005760  116262  000126  000127             MOVB    BCC(R2),BCL(R2)         ;SET THE BCC LATCH
2542                                            .IF DF LS.11
2543                                            MOVB    CBAX,CRCCHR             ;ACCUMLATE END CHAR IN BCC
2544                                            JSR     PC,$CRC16               ;
2545                                            .IFF
2546 005766  110537  170704                     MOVB    CBAX,@#KGDATA           ;ACCUMULATE END CHAR IN BCC
2547                                            .ENDC
2548                                            .IF DF RSTS
2549                                    1$:     TSTB    @#KGSTAT                ;DONE YET?
2550                                            BPL     1$                      ;NO, BUT IT WILL SOON. .
2551                                            .ENDC
2552                                            .IF DF LS111
2553                                            MOV     CRCACC,CBC(R2)          ;STORE COMPUTED BCC
2554                                            .IFF
2555 005772  013762  170702  000122             MOV     @#KGBCCR,CBC(R2)        ;STORE COMPUTED BCC
2556                                            .ENDC
2557 006000  000754                             BR      GTGET
2558
2559                                            ;PROCESS ETX
2560 006002  052762  010000  000012  GTETX:    BIS     #MEND,RSW(R2)           ;SET 'MESSAGE END' FLAG
2561 006010  000755                             BR      GTETO
2562
2563                                            ; PROCESS ITB
2564 006012                             GTITB:
2565 006012  052762  002000  000010  GTITB1:   BIS     #ITBA,LSW(R2)           ;SET ITB ACTIVE
2566 006020  032762  100000  000006             BIT #RCVT,LCW(R2)               ;IF TRANSPARENT...
2567 006026  001751                             BEQ GTET1
2568 006030  042762  100000  000010             BIC #MIP,LSW(R2)                ;...LET DLE STX SET MIP AGAIN
2569 006036  000745                             BR GTET1
2570
2571
2572                                            ;PROCESS ENQ (GTENQ) WITHIN A RECORD
```

```
2573 006040  032762  100000  000012  GTENQ:  BIT #ARF,RSW(R2)
2574 006046  001003                  BNE GTENQ1              ;IF ARF SET, DON'T GET EXCITED ABOUT ENQ
2575
2576 006050  052762  040000  000012  GTEOT:  BIS #AEND,RSW(R2)              ;SET 'ABNORMAL END'FLAG
2577 006056  042762  102200  000010  GTENQ1: BIC #MIP!BCCA!ITBA,LSW(R2)     ;CLEAR PROCESS FLAGS
2578 006064  000167  176750            JMP ARFENC
2579
2580
2581                                  ;PROCESS SYN WITHIN A RECORD
2582
2583 006070  032762  000400  000010  GTSYN:  BIT     #SRTA,LSW(R2)   ;IS SLAVE RCVE TIMER ACTIVE
2584 006076  001715                  BEQ     GTGET           ;IGNORE SYN AND EXIT
2585 006100  004767  176534          JSR     PC,SSRTM        ;START NEW TIMER, CLEARS ANY PREVIOUS TIMERS
2586 006104  000712                  BR      GTGET           ;EXIT
2587
```

```
2589 006106  105362  000127          BCCHK:  DECB BCL(R2)             ;INCREMENT THE LATCH
2590 006112  001411                          BEQ GTBC1               ;IF ZERO, GO HOUSEKEEP
2591 006114  126204  000122                  CMPB CBC(R2),CBBX               ;IF NOT, CHECK CURRENT BCC CHAR
2592 006120  001403                          BEQ GTBC2               ;OK
2593 006122  052762  000004  000012          BIS #DCH,RSW(R2)        ;DOESN'T MATCH - SE  D TA CHECK FLAG
2594 006130  000362  000122          GTBC2:  SWAB CBC(R2)            ;SET FOR NEXT PASS
2595 006134  000676                          BR GTGET
2596
2597 006136  032762  002000  000010  GTBC1:  BIT #ITBA,LSW(R2)                ;ITB ACTIVE?
2598 006144  001004                          BNE GTIT1                       ;YES
2599 006146  042762  100200  000010          BIC     #MIP!BCCA,LSW(R2)
2600 006154  000436                          BR      INIEN1                  ;MUST HAVE BEEN A PAD -IGNORE IT
2601                                                                         ;AND GO MAKE UIB ENTRY
2602
2603 006156  010446          GTIT1:  MOV CBBX,-(SP)                  ;SAVE CHAR FOR LATER
2604 006160  004767  171740                  JSR PC,C.BRCD                   ;WHILE WE MAKE UIB ENTRY
2605 006164  012604                          MOV (SP)+,CBBX          ;NOW PROCESS THE CHAR
2606 006166  005062  000122                  CLR CBC(R2)                     ;CLEAR OUT OLD BCC FROM LINE TABLE
2607                                         .IF DF LSI11
2608                                         CLR     CRCACC          ;CLEAR THE BCC ACCUMULATOR
2609                                         .IFF
2610 006172  052737  000020  170700          BIS #20,@#KGSTAT        ;CLEAR THE KG11 BCC REGISTER
2611                                         .ENDC
2612 006200  016201  000132                  MOV CCT(R2),R1          ;RESTORE CCT POINTER
2613 006204  000167  175766                  JMP GET2
2614
2615
2616 006210  032762  000020  000006  INIENQ: BIT #MSS,LCW(R2)          ;WHICH MODE AR WE IN?
2617 006216  001415                          BEQ INIEN1              ;MASTER - GO TRY AGAIN
2618
2619                                 ;ENQ CONTENTION
2620 006220  032762  000004  000006  ENQAGN: BIT #MCF,LCW(R2)          ;IS THIS THE SECOND ENQ?
2621 006226  001413                          BEQ ENQAG1              ;NO - GO SET TIMER FOR RETRY
2622 006230  042762  000024  000006          BIC #MCF!MSS,LCW(R2)    ;YES - GO INTO SLAVE MODE
2623 006236  052762  000200  000014          BIS #CMXM,TSW(R2)       ;AND NOTIFY THE USER TO DO SAME
2624                                         .IF NDF RSTS
2625 006244  005000                          CLR     R0              ;SET UP LINE NO
2626 006246  004767  000532'                 JSR PC,TDENT            ;INDICATE TRANSMIT OVERRIDE
2627                                         .IFF
2628                                         JSR PC,TDENTO           ;INDICATE TRANSMIT OVERRIDE
2629                                         .ENDC
2630 006252  000167  176562          INIEN1: JMP     ARFEND
2631
2632                                 ; TRY AGAIN WITH ENQ XMIT AFTER 1 OR 3 SECONDS
2633 006256  052762  000004  000006  ENQAG1: BIS #MCF,LCW(R2)          ;SET CONTENTION FLAG
2634 006264  004767  000010                  JSR PC,ENQSET           ;GO SET THE TIMER
2635 006270  004767  000230'         ENQZ:   JSR PC,ARRCV1           ;GO RESET LINE
2636 006274  000167  176560                  JMP ARF1                ;AND FLUSH TRAILING CHARS
2637
2638                                 ; ENQ TIMER SETUP ROUTINE - USED HERE AND BY PUTD ROUTINE
2639 006300                          ENQSET:
2640 006300  005000                          CLR     R0              ;RECREATE LINE NUMBER
2641                                         .IF DF  UAPTIM
2642 006302  016704  171472                  MOV     CONTIM,R4       ;SET TIMEOUT VALUE FOR CONTENTION
2643                                         .IFF
2644                                         MOV #CONTIM,R4          ;SET CONTENTION TIMEOUT
2645                                         .ENDC
```

```
2646 006306  012705  006322'                MOV #ENQTIM,R5          ;ROUTINE POINTER
2647 006312  004767  000000G                JSR     PC,CTDEL        ;DELETE ANY EXISTING BID TIMERS
2648 006316  000167  000000G                JMP     CTIME           ;
2649
2650                                 ;COME HERE AFTER CONTENTION TIMEOUT PERIOD
2651 006322  032762  002000  000014 ENQTIM: BIT #BSNT,TSW(R2)       ;BID COMPLETED ALREADY?
2652 006330  001410                        BEQ ENQY                ;YES - IGNORE THE TIMER
2653 006332  004767  177742                JSR PC,ENQSET           ;SET TIMER AGAIN
2654 006336  012700  000000                MOV #C.BTSC,R0          ;AND SEND THE ENQ AGAIN
2655 006342  016201  000130                MOV TCP(R2),R1          ;XMIT CHAIN POINTER
2656 006346  004767  000000G                JSR PC,C.SARP           ;XMIT STARTUP
2657 006352  000207                 ENQY:  RTS PC          ;EXIT TO TIMER
```

```
2659                                    .IF DF RSTS
2660                            ;       ROUTINE TO CALCULATE THE BUFFER ADDRESS FOR RSTS
2661                            ;       WHEN THE BUFFERS CAN BE MAPPED OUTSIDE OF THE
2662                            ;       MONITOR AREA
2663                            ;       THE ROUTINE IS ENTERED WITH THE ADD IN R5
2664                            ;       AND RETURNS WITH THE MAPPED ADD IN R5
2665                                    .GLOBL  SAVR6,SAVADD
2666                            GTBF1:  MOV     R5,-(SP)        ;SAVE OFF R5
2667                                    MOV     ARA(R2),R5              ;PUT ADDRESS IN R5
2668                                    JSR     PC,GTBFAD               ;DO MAPPING
2669                                    MOV     R5,SAVADD               ;SAVE OFF ADDRESS
2670                                    MOV     (SP)+,R5                ;RESTORE R5
2671                                    RTS     PC
2672                            ;
2673                            GTBFFD:
2674                                    MOV     R4,-(SP)        ;SAVE OFF REGISTERS
2675                                    MOV     R3,-(SP)        ;
2676                                    MOV     R5,R4           ;GET ADDRESS
2677                                    CALLX   MAPBUF          ;GET MAPPED ADDRESS
2678                                    MOV     R3,R5           ;
2679                                    ADD     #BUFHDR,R5      ;STEP OVER BUFFER HEADER
2680                                                            ;WHERE EXACT SIZE IS KEPT
2681                                    MOV     KISAR6,SAVR6    ;SAVE OFF APR 6 JUST IN CASE
2682                                    MOV     (SP)+,R3        ;RESTORE REGISTERS
2683                                    MOV     (SP)+,R4        ;
2684                                    RTS     PC              ;RETURN
2685                            ;
2686                                    .ENDC
2687                                    .ENDC
2688                                    .ENDC
2689                                    .ENDC
2690            000001                  .END
```

BISYNC MACRO M1104 06-DEC-77 08:18 PAGE 45-1
SYMBOL TABLE

```
AAF    = 000004          CLC    = 000241          C.B0    000376R     003 GTETB   005736R     002 MSRT1   004464R     002
ABC    = 000024          CLN    = 000250          C.B1    000404R     003 GTETX   006002R     002 MSS    = 000020
ACKOUT  000150R     003  CLNV   = 000252          C.B5    000530R     003 GTETO   005744R     002 MTOUT  = 000100
ACTTAB  003652R     002  CLV    = 000242          C.SARP= ****** G         GTET1   005752R     002 NAK    = 000025
AEF    = 020000          CLZ    = 000244          C.SFY = ****** G         GTFRST  005170R     002 NAKCNT= 000010
AEND   = 040000          CNAK   = 000024          C.STAT= 177776           GTFRT1  005206R     002 NAKF   = 100000
AEXB    003452R     002  CONTIM  000000R     002 C1     = 000000          GTGET   005732R     002 NAT    = 001000
AFX    = 002000          COVR   = 000040          C2     = 000007          GTITB   006012R     002 NEF    = 004000
AKT    = 000136          CPAD   = 000002          DCH    = 000004          GTITB1  006012R     002 NMP    = 000044
AMP    = 000042          CPD26  = 000001          DLE    = 000020          GTIT1   006156R     002 NOP    = 000240
ANF    = 040000          CRBINT= 000000           DLEF   = 000200          GTNONE  004736R     002 NUB    = 000002
ARA    = 000022          CRBSRV  004022R     002 DSC    = 004000          GTNOPE  005120R     002 OOF    = 000100
ARBSET  000130R     003  CRD    = 000001          EBCACK  000010R     003 GTNO1   004744R     002 OS     = 000000
ARBSP   000054R     003  CRVI   = 000000          EBCCCT  000000R     003 GTNO2   004760R     002 OVXM   = 000200
ARB1    000116R     003  CSA    = 000002          EBCNAK  000024R     003 GTNO3   005012R     002 PAD    = 000177
ARF    = 100000          CSAT    000262R     003 EBCRTR  004006R     002 GTNSAR  004370R     002 PARITY= 010000
ARFEND  005040R     002  CSOH   = 000004          EBCSYN  000016R     003 GTSAT   003712R     002 PCH    = 000010
ARF1    005060R     002  CSSRT   004422R     002 EBCXTR  001752R     002 GTSTX   005710R     002 PPD    = 000046
ARRCV   000212R     003  CSTX   = 000003          EBCXT1  001770R     002 GTSYN   006070R     002 PTBSP   000746R     002
ARRCV1  000230R     003  CSYN   = 000016          ECH    = 000121          GTUOF   004374R     002 PTCON   002326R     002
ARSEXT  000172R     003  CTDEL  = ****** G        EKT    = 000140          HALT   = 000000          PTC1    002046R     002
ASCCCT= 000000           CTIME  = ****** G        ENQ    = 000005          INI    = 000400          PTC2    002056R     002
ASCI   = 000000          CWAK   = 000001          ENQAGN  006220R     002 INIENQ  006210R     002 PTC3    002346R     002
ASSGN1  003010R     002  C.BAB1  002762R     002 ENQAG1  006256R     002 INIEN1  006252R     002 PTC4    002102R     002
ASSGN2  003036R     002  C.BARB  002704RG    002 ENQCNT= 000010           INTBCC  005340R     002 PTDAB   001730R     002
ASSGN3  003050R     002  C.BARX  002756R     002 ENQSET  006300R     002 INTC1   004676R     002 PTDLE   000756R     002
ASX    = 000016          C.BAR1  002714R     002 ENQTIM  006322R     002 INTC2   004716R     002 PTDL1   001006R     002
ATR    = 000040          C.BCR1  000352R     002 ENQY    006352R     002 INTTIC  004656R     002 PTDZ    001354R     002
ATT    = 000036          C.BCR2  000366R     002 ENQZ    006270R     002 INTVTM  004610R     002 PTENQ   001176R     002
BCC    = 000126          C.BCR3  000262R     002 EOT    = 000004          INTVT1  004632R     002 PTENQ1  001226R     002
BCCA   = 000200          C.BEOF= 002000           ESNT   = 001000          IPL    = 000002          PTEOT   001244R     002
BCCHK   006106R     002  C.BGBC= 000003          ETA     003052RG    002 IRC    = 000062          PTEXIT  002454R     002
BCL    = 000127          C.BGMS= 000105           ETB    = 000027          IRS    = 000036          PTEXN   002450R     002
BEND   = 020000          C.BGRD= 000303           ETX    = 000003          ISC    = 000072          PTEXT   002424R     002
BENQ    004532R     002  C.BGTD= 000304           ETXF   = 000400          ITB    = 000037          PTEX1   002406R     002
BSIZE  = 000620          C.BIMS  002004R     002 FBF    = 000400          ITBA   = 002000          PTEX3   002406R     002
BSNT   = 002000          C.BINT  000020RG    002 FRSAT   003736R     002 ITS    = 004000          PTITB   001012R     002
BTCE   = 000027          C.BLTL= 000776 G         GENQ10  005154R     002 KGBCCR= 170702           PTIT1   001064R     002
BTCI   = 000030          C.BMDC= 000002           GETCHA  004076R     002 KGC    = 000034          PTIT2   001162R     002
CAK    = 000134          C.BMMC= 000001           GETCHR  004064R     002 KGDATA= 170704           PTIT3   001040R     002
CAK0   = 000052          C.BMSE  000370RG    002 GETCH1  005410R     002 KGSTAT= 170700           PTIT4   001124R     002
CAK1   = 000053          C.BNXT  000312R     002 GET1    004106R     002 LASBYT= 000031           PTIT4A  001120R     002
CA.ASC= 000000           C.BODE  000316RG    003 GET1A   004126R     002 LCW    = 000006          PTMODE  002264R     002
CA.EBC= 000001           C.BOD1  000356R     003 GET1B   004150R     002 LDGRPH= 000007           PTMORS  002456R     002
CA.NCR= 000001           C.BPD   000372RG    002 GET2    004176R     002 LGC    = 000152          PTM1    002602R     002
CA.SBT= 000000           C.BPDR  000404R     002 GET3    004324R     002 LSW    = 000010          PTM2    002574R     002
CBAX  =%000005           C.BPDX  000414R     002 GET4    004246R     002 LTL    = 000776          PTNEXT  000616R     002
CBBX  =%000004           C.BPTC  002014RG    002 GET5    004216R     002 MBX    = 000034          PTNORM  000534R     002
CBC    = 000122          C.BRCD  000124R     002 GET9    004306R     002 MCF    = 000004          PTNO    000600R     002
CCC    = 000257          C.BRCY  000220R     002 GNX1    004362R     002 MDC    = 000050          PTN1    000620R     002
CCT    = 000132          C.BRCZ  000206R     002 GOUOF   004412R     002 MEND   = 010000          PTR1    000630R     002
CDLE   = 000010          C.BRMC  000010R     002 GTBCC   004232R     002 MIP    = 100000          PTR2    000660R     002
CENQ   = 000022          C.BTSC= 000000           GTBC1   006136R     002 MRC    = 000115          PTR3    000634R     002
CEOT   = 000026          C.BUD2  000424R     003 GTBC2   006130R     002 MSRT    000002R     002 PTR4    000676R     002
CETB   = 000005          C.BUD3  000464R     003 GTENQ   006040R     002 MSRTO   004436R     002 PTSAT   003762R     002
CETX   = 000006          C.BUD4  000512R     003 GTENQ1  006056R     002 MSRTOA  004514R     002 PTSLVE  002430R     002
CITB   = 000007          C.BUFF= 000007           GTEOT   006050R     002 MSRTOX  004530R     002 PTSTX   001166R     002
```

```
BISYNC  MACRO M1104  06-DEC-77 08:18  PAGE 45-2
SYMBOL TABLE

PTSYN   001272R     002 RCVT   = 100000          SETENQ  005652R     002 STX    = 000002          TRYNAK  000174R     003
PTS1    002440R     002 RCVTMT= ****** G         SETEOF  005554R     002 STXBCC  005726R     002 TSA    = 000066
PTTERM  002620R     002 RCX    = 000020          SETEOT  005424R     002 STXF   = 000020          TSW    = 000014
PTTEXT  002660R     002 RDBF   = 001000          SETETB  002176R     002 SYN    = 000026          TTM    = 000110
PTT1    002642R     002 REPEAT  005146R     002 SETJ    005706R     002 SYP    = 000032          TT2A   = 000144
PUTD0   001310R     002 RETNX =%000007           SETMIP  005274R     002 TBC    = 000070          TT2C   = 000146
PUTD1   000512R     002 RIB    = 000137          SETM1   005404R     002 TBS    = 000112          TT3A   = 000150
PUTD1A  000422R     002 RJT    = 000040          SETNAK  005446R     002 TCI3    002112R     002 TXC    = 000064
PUTD10  001720R     002 RKT    = 000141          SETN1   005470R     002 TCJ1    002506R     002 T1A    = 000074
PUTD11  000420R     002 ROVR   = 000100          SETRVI  005544R     002 TCJ2    002532R     002 T1C    = 000076
PUTD22  001462R     002 RSW    = 000012          SETSYN  002136R     002 TCJ3    002556R     002 T2A    = 000100
PUTD23  001430R     002 RSX    = 000001          SETWAK  005160R     002 TCP    = 000130          T2C    = 000102
PUTD24  001452R     002 RVI    = 040000          SETWRG  005622R     002 TCT    = 000120          T3A    = 000104
PUTD3   001370R     002 R6     =%000006          SEV    = 000262          TCW    = 000142          T3C    = 000106
PUTD4   001516R     002 R7     =%000007          SEZ    = 000264          TDENT   000532R     003 UAPTIM= 000000
PUTD5   001540R     002 SBTCCT= 000000           SIS    = 000010          THALT  = 000240          UBL    = 000026
PUTD5A  001544R     002 SCC    = 000277          SOH    = 000001          THC    = 000116          UOVR   = 000020
PUTD6   001602R     002 SEC    = 000261          SRC    = 000153          TIB    = 000154          XAC    = 010000
PUTD7   001636R     002 SEN    = 000270          SRTA   = 000400          TICINT  000006R     002 XMTTMT= ****** G
PUTD8   001644R     002 SENV   = 000272          SRTOF  = 000200          TRANER  005476R     002 XPT    = 020000
PUTD9   001710R     002 SETACK  005524R     002 SSRT    000004R     002 TRAN1   005504R     002 XRC    = 000001
RBA    = 000001         SETAC1  005532R     002 SSRTM   004640R     002 TRAN2   005514R     002 XXP    = 010000
RBC    = 000124         SETARF  005262R     002 SSRTO   004720R     002 TRC    = 000114          X.     = 000002
RBU    = 000004         SETDLE  005414R     002 STARF1  005640R     002 TRNREC= 001000           X1      001776R     002
RCVSIB= 000000

. ABS.  000000    000
        000000    001
XXX     006354    002
ISR     000624    003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:   2238 WORDS  ( 9 PAGES)
DYNAMIC MEMORY:   2810 WORDS  ( 10 PAGES)
ELAPSED TIME:   00:01:02
 256,020 BISYNC;27, 256,030 BISYNC/CRF/-SP=BIRXM,OPDEF,BISYNC
```

BISYNC CREATED BY MACRO ON 6-DEC-77 AT 08:18 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE	REFERENCES								
AAF	= 000004	#7-466	15-741	24-1367	42-2510					
ABC	= 000024	#6-398	6-399	15-704	*15-720	*24-1335	37-2105	*37-2131		
ACKOUT	000150 R	19-1028	#24-1364	40-2205	41-2349					
ACTTAB	003652 R	18-853	#36-1977	38-2135	42-2367					
AEF	= 020000	#7-476	16-778	16-791	18-915	18-917	25-1427	25-1438	26-1474	39-2171
		41-2339								
AEND	= 040000	#8-497	43-2576							
AEXB	003452 R	20-1072	#35-1801							
AFX	= 002000	#7-457	16-779	16-790	18-859	18-911	18-922	19-956	19-999	
AKT	= 000136	#6-425	6-426	6-427	*21-1142	24-1356	*24-1360	*25-1435	*42-2484	
AMP	= 000042	#9-542	9-543	24-1355						
ANF	= 040000	#7-477	16-778	18-915	18-917	25-1425	26-1474	37-2107		
ARA	= 000022	#6-397	6-398	*15-716	*24-1331	37-2114	*37-2130			
ARBSET	000130 R	#24-1361	24-1374							
ARBSP	000054 R	24-1342	#24-1349							
ARB1	000116 R	24-1357	#24-1359							
ARF	= 100000	#8-498	42-2439	43-2573						
ARFEND	005040 R	#41-2319	42-2440	42-2486	43-2578	44-2630				
ARF1	005060 R	41-2321	#41-2323	41-2350	41-2361	42-2466	44-2636			
ARRCV	000212 R	24-1351	#24-1378	25-1430	42-2455	42-2465				
ARRCV1	000230 R	#24-1382	44-2635							
ARSEXT	000172 R	#24-1369	24-1379	24-1381	24-1392					
ASCCCT	= 000000	#10-556								
ASCI	= 000000	#3-238	19-1046	19-1051	20-1070	35-1782				
ASSGN1	003010 R	24-1327	#24-1329							
ASSGN2	003036 R	24-1330	#24-1337							
ASSGN3	003050 R	24-1320	#24-1344							
ASX	= 000016	#6-395	6-396	24-1300	*24-1325	24-1326	*24-1328			
ATR	= 000040	#9-541	9-542	37-2088						
ATT	= 000036	#9-540	9-541	19-992						
BCC	= 000126	#6-420	6-421	6-422	*21-1132	43-2541				
BCCA	= 000200	#7-470	37-2095	41-2324	42-2393	43-2577	44-2599			
BCCHK	006106 R	37-2087	#44-2589							
BCL	= 000127	#6-421	24-1380	37-2085	41-2280	*43-2541	*44-2589			
BEND	= 020000	#8-496	43-2538							
BENQ	004532 R	40-2204	#40-2211	41-2348						
BSIZE	= 000620	#4-339	6-440	17-821	18-886					
BSNT	= 002000	#8-509	18-918	19-1021	25-1449	42-2478	44-2651			
BTCE	= 000027	#9-530	9-531	19-1013						
BTCI	= 000030	#9-531	9-536	21-1132						
CAK	= 000134	#6-424	6-425	21-1128	*21-1133	*21-1138	*25-1434	42-2378	42-2380	*42-2471
		*42-2485								
CAK0	= 000052	#9-546	9-547	21-1133	21-1136	25-1434	42-2485			
CAK1	= 000053	#9-547								
CA.ASC	= 000000	#4-291								
CA.EBC	= 000001	#4-292	11-564	11-601	20-1059	27-1484	37-2030			
CA.NCR	= 000001	#4-294								
CA.SBT	= 000000	#4-293								
CBAX	=%000005	#3-268	*37-2033	37-2101	37-2112	42-2378	42-2380	42-2382	42-2384	42-2430
		43-2533	43-2546							
CBBX	=%000004	#3-269	*37-2032	37-2033	37-2034	*37-2034	37-2074	*37-2076	37-2092	*37-2112
		37-2114	38-2135	38-2140	42-2365	42-2367	42-2369	44-2591	44-2603	*44-2605

SYMBOL		VALUE		REFERENCES								
CBC	=	000122		#6-418	6-419	17-840	*18-882	*19-1018	37-2050	*41-2285	*42-2401	*43-2555
				44-2591	*44-2594	*44-2606						
CCC	=	000257		#2-10								
CCT	=	000132		#6-423	6-424	16-795	21-1127	*21-1129	24-1349	24-1385	25-1433	37-2051
				40-2203	41-2323	44-2612						
CDLE	=	000010		#9-525	9-526	18-861	18-876	19-973	19-1003	42-2420	43-2523	
CENQ	=	000022		#9-527	9-528	40-2222						
CEOT	=	000026		#9-529	9-530							
CETB	=	000005		#9-522	9-523	16-796	17-842	21-1139				
CETX	=	000006		#9-523	9-524	16-799						
CITB	=	000007		#9-524	9-525	18-889						
CLC	=	000241		#2-6								
CLN	=	000250		#2-9	3-278							
CLNV	=	000252		#3-278	24-1343							
CLV	=	000242		#2-7	3-278							
CLZ	=	000244		#2-8								
CNAK	=	000024		#9-528	9-529							
CONTIM		000000	R	#4-325	*15-674	44-2642						
COVR	=	000040		#8-487	37-2073							
CPAD	=	000002		#9-519	9-520	19-1027	19-1048	24-1362				
CPD26	=	000001		#3-241								
CRBINT	=	000000		#1-2	4-368	37-2062	37-2066	41-2290	41-2327	41-2331		
CRBSRV		004022	R	24-1396	#37-2042	41-2296						
CRD	=	000001		#8-482								
CRVI	=	000000		#9-517	9-518	42-2384						
CSA	=	000002		#7-465	24-1391	24-1393	41-2291					
CSAT		000262	R	22-1194	#24-1389							
CSOH	=	000004		#9-521	9-522							
CSSRT		004422	R	15-683	#38-2156	41-2322						
CSTX	=	000003		#9-520	9-521	19-975						
CSYN	=	000016		#9-526	9-527	19-965	21-1131					
CTDEL	=	******	G	3-246	44-2647							
CTIME	=	******	G	3-246	24-1397	40-2239	41-2297	44-2648				
CWAK	=	000001		#9-518	9-519	42-2382						
C.BAB1		002762	R	24-1311	#24-1323							
C.BARB		002704	RG	3-248	#24-1294							
C.BARX		002756	R	24-1299	#24-1319							
C.BAR1		002714	R	#24-1298								
C.BCR1		000352	R	15-738	#15-743							
C.BCR2		000366	R	15-744	#15-746							
C.BCR3		000262	R	15-723	#15-728							
C.BEOF	=	002000		#8-493	42-2483							
C.BGBC	=	000003		#13-613	13-615	13-616	13-617					
C.BGMS	=	000105		#13-617								
C.BGRD	=	000303		#13-615	15-734							
C.BGTD	=	000304		#13-616	26-1470							
C.BIMS		002004	R	*15-724	15-726	#21-1097	*21-1131	*21-1145	21-1150	21-1154	*21-1160	21-1165
				*24-1254	24-1255	*24-1383	*24-1386	24-1387				
C.BINT		000020	RG	3-248	#15-657							
C.BLTL	=	000776	G	3-249	#6-441							
C.BMDC	=	000002		#13-625	15-725	21-1162	24-1384					
C.BMMC	=	000001		#13-624	21-1147	24-1253						

BISYNC CREATED BY MACRO ON 6-DEC-77 AT 08:18 PAGE 3
SYMBOL CROSS REFERENCE CREF

SYMBOL		VALUE		REFERENCES							
C.BMSE		000370	RG	3-254	#16-754						
C.BNXT		000312	R	15-730	#15-733						
C.BODE		000316	RG	3-254	#25-1412						
C.BOD1		000356	R	25-1413	#25-1419						
C.BPD		000372	RG	3-248	#16-765						
C.BPDR		000404	R	#16-770	16-807	17-832					
C.BPDX		000414	R	#16-772	16-777						
C.BPTC		002014	RG	3-248	#21-1106						
C.BRCD		000124	R	#15-698	24-1260	41-2319	44-2604				
C.BRCY		000220	R	15-714	#15-716						
C.BRCZ		000206	R	15-709	#15-713						
C.BRMC		000010	R	#14-632	*15-701	*15-703	*15-704	*15-728	*15-731	*15-733	*15-734	15-735
C.BTSC	=	000000		#13-623	22-1190	24-1365	44-2654					
C.BUD2		000424	R	25-1424	25-1426	#25-1430						
C.BUD3		000464	R	25-1432	#25-1438							
C.BUD4		000512	R	25-1439	25-1441	#25-1449						
C.BUFF	=	000007		#4-335	4-336	4-336	4-336	4-336	18-884			
C.B0		000376	R	#25-1423	25-1428							
C.B1		000404	R	25-1420	25-1422	#25-1425						
C.B5		000530	R	25-1450	#25-1453							
C.SARP	=	******	G	3-247	15-727	21-1152	21-1167	22-1192	24-1256	24-1366	24-1388	44-2656
C.SFY	=	******	G	3-247	15-736	26-1472						
C.STAT	=	177776		#3-257	24-1307	24-1308	24-1316	24-1340				
C1	=	000000		#35-1804	35-1804	35-1804	#35-1805	35-1805	35-1805	#35-1806	35-1806	35-1806
				#35-1807	35-1807	35-1807	#35-1808	35-1808	35-1808	#35-1809	35-1809	35-1809
				#35-1810	35-1810	35-1810	#35-1811	35-1811	35-1811	#35-1814	35-1814	35-1814
				#35-1815	35-1815	35-1815	#35-1816	35-1816	35-1816	#35-1817	35-1817	35-1817
				#35-1818	35-1818	35-1818	#35-1819	35-1819	35-1819	#35-1820	35-1820	35-1820
				#35-1821	35-1821	35-1821	#35-1824	35-1824	35-1824	#35-1825	35-1825	35-1825
				#35-1826	35-1826	35-1826	#35-1827	35-1827	35-1827	#35-1829	35-1829	35-1829
				#35-1830	35-1830	35-1830	#35-1831	35-1831	35-1831	#35-1832	35-1832	35-1832
				#35-1835	35-1835	35-1835	#35-1836	35-1836	35-1836	#35-1837	35-1837	35-1837
				#35-1838	35-1838	35-1838	#35-1839	35-1839	35-1839	#35-1840	35-1840	35-1840
				#35-1841	35-1841	35-1841	#35-1842	35-1842	35-1842	#35-1845	35-1845	35-1845
				#35-1846	35-1846	35-1846	#35-1847	35-1847	35-1847	#35-1848	35-1848	35-1848
				#35-1849	35-1849	35-1849	#35-1850	35-1850	35-1850	#35-1851	35-1851	35-1851
				#35-1852	35-1852	35-1852	#35-1855	35-1855	35-1855	#35-1856	35-1856	35-1856
				#35-1857	35-1857	35-1857	#35-1858	35-1858	35-1858	#35-1859	35-1859	35-1859
				#35-1860	35-1860	35-1860	#35-1861	35-1861	35-1861	#35-1862	35-1862	35-1862
				#35-1865	35-1865	35-1865	#35-1865	35-1865	#35-1866	35-1866	35-1866	#35-1866
				35-1866	#35-1867	35-1867	35-1867	#35-1867	35-1867	#35-1868	35-1868	35-1868
				#35-1868	35-1868	#35-1869	35-1869	35-1869	#35-1869	35-1869	#35-1870	35-1870
				35-1870	#35-1870	35-1870	#35-1871	35-1871	35-1871	#35-1871	35-1871	#35-1872
				35-1872	35-1872	#35-1872	35-1872	#35-1875	35-1875	35-1875	#35-1875	35-1875
				#35-1876	35-1876	35-1876	#35-1876	35-1876	#35-1877	35-1877	35-1877	#35-1878
				35-1878	35-1878	#35-1879	35-1879	35-1879	#35-1880	35-1880	35-1880	#35-1881
				35-1881	35-1881	#35-1882	35-1882	35-1882	#35-1887	35-1887	35-1887	#35-1888
				35-1888	35-1888	#35-1888	35-1888	#35-1889	35-1889	35-1889	#35-1889	35-1889
				#35-1890	35-1890	35-1890	#35-1890	35-1890	#35-1891	35-1891	35-1891	#35-1891
				35-1891	#35-1892	35-1892	35-1892	#35-1892	35-1892	#35-1893	35-1893	35-1893
				#35-1893	35-1893	#35-1894	35-1894	35-1894	#35-1894	35-1894	#35-1896	35-1896
				35-1896	#35-1896	35-1896	#35-1897	35-1897	35-1897	#35-1897	35-1897	#35-1898

SYMBOL VALUE REFERENCES

35-1898 35-1898 #35-1898 35-1898 #35-1899 35-1899 35-1899 #35-1899 35-1899
#35-1900 35-1900 35-1900 #35-1900 35-1900 #35-1901 35-1901 35-1901 #35-1901
35-1901 #35-1902 35-1902 35-1902 #35-1902 35-1902 #35-1903 35-1903 35-1903
#35-1903 35-1903 #35-1906 35-1906 35-1906 #35-1906 35-1906 #35-1907 35-1907
35-1907 #35-1907 35-1907 #35-1908 35-1908 35-1908 #35-1908 35-1908 #35-1909
35-1909 35-1909 #35-1909 35-1909 #35-1910 35-1910 35-1910 #35-1910 35-1910
#35-1911 35-1911 35-1911 #35-1911 35-1911 #35-1912 35-1912 35-1912 #35-1912
35-1912 #35-1913 35-1913 35-1913 #35-1913 35-1913 #35-1916 35-1916 35-1916
#35-1916 35-1916 #35-1917 35-1917 35-1917 #35-1917 35-1917 #35-1918 35-1918
35-1918 #35-1918 35-1918 #35-1919 35-1919 35-1919 #35-1920 35-1920 35-1920
#35-1920 35-1920 #35-1921 35-1921 35-1921 #35-1922 35-1922 35-1922 #35-1923
35-1923 35-1923 #35-1929 35-1929 35-1929 #35-1930 35-1930 35-1930 #35-1931
35-1931 35-1931 #35-1932 35-1932 35-1932 #35-1933 35-1933 35-1933 #35-1934
35-1934 35-1934 #35-1935 35-1935 35-1935 #35-1936 35-1936 35-1936 #35-1939
35-1939 35-1939 #35-1940 35-1940 35-1940 #35-1941 35-1941 35-1941 #35-1942
35-1942 35-1942 #35-1943 35-1943 35-1943 #35-1944 35-1944 35-1944 #35-1945
35-1945 35-1945 #35-1946 35-1946 35-1946 #35-1949 35-1949 35-1949 #35-1950
35-1950 35-1950 #35-1951 35-1951 35-1951 #35-1952 35-1952 35-1952 #35-1952
35-1952 #35-1953 35-1953 35-1953 #35-1953 35-1953 #35-1954 35-1954 35-1954
#35-1954 35-1954 #35-1955 35-1955 35-1955 #35-1955 35-1955 #35-1956 35-1956
35-1956 #35-1956 35-1956 #35-1959 35-1959 35-1959 #35-1959 35-1959 #35-1960
35-1960 35-1960 #35-1960 35-1960 #35-1961 35-1961 35-1961 #35-1961 35-1961
#35-1962 35-1962 35-1962 #35-1962 35-1962 #35-1963 35-1963 35-1963 #35-1964
35-1964 35-1964 #35-1964 35-1964 #35-1965 35-1965 35-1965 #35-1965 35-1965
#35-1966 35-1966 35-1966

C2 = 000007 #35-1804 35-1804 35-1804 #35-1805 35-1805 35-1805 #35-1806 35-1806 35-1806
#35-1807 35-1807 35-1807 #35-1808 35-1808 35-1808 #35-1809 35-1809 35-1809
#35-1809 35-1809 #35-1810 35-1810 35-1810 #35-1810 35-1810 #35-1811 35-1811
35-1811 #35-1811 35-1811 #35-1814 35-1814 35-1814 #35-1815 35-1815 35-1815
#35-1816 35-1816 35-1816 #35-1817 35-1817 35-1817 #35-1817 35-1817 #35-1818
35-1818 35-1818 #35-1818 35-1818 #35-1819 35-1819 35-1819 #35-1819 35-1819
#35-1820 35-1820 35-1820 #35-1820 35-1820 #35-1821 35-1821 35-1821 #35-1821
35-1821 #35-1824 35-1824 35-1824 #35-1825 35-1825 35-1825 #35-1826 35-1826
35-1826 #35-1827 35-1827 35-1827 #35-1829 35-1829 35-1829 #35-1829 35-1829
#35-1830 35-1830 35-1830 #35-1830 35-1830 #35-1831 35-1831 35-1831 #35-1832
35-1832 35-1832 #35-1835 35-1835 35-1835 #35-1836 35-1836 35-1836 #35-1837
35-1837 35-1837 #35-1837 35-1837 #35-1838 35-1838 35-1838 #35-1839 35-1839
35-1839 #35-1840 35-1840 35-1840 #35-1841 35-1841 35-1841 #35-1842 35-1842
35-1842 #35-1842 35-1842 #35-1845 35-1845 35-1845 #35-1846 35-1846 35-1846
#35-1846 35-1846 #35-1847 35-1847 35-1847 #35-1847 35-1847 #35-1848 35-1848
35-1848 #35-1848 35-1848 #35-1849 35-1849 35-1849 #35-1849 35-1849 #35-1850
35-1850 35-1850 #35-1850 35-1850 #35-1851 35-1851 35-1851 #35-1852 35-1852
35-1852 #35-1852 35-1852 #35-1855 35-1855 35-1855 #35-1855 35-1855 #35-1856
35-1856 35-1856 #35-1856 35-1856 #35-1857 35-1857 35-1857 #35-1857 35-1857
#35-1858 35-1858 35-1858 #35-1858 35-1858 #35-1859 35-1859 35-1859 #35-1859
35-1859 #35-1860 35-1860 35-1860 #35-1861 35-1861 35-1861 #35-1861 35-1861
#35-1862 35-1862 35-1862 #35-1865 35-1865 35-1865 #35-1866 35-1866 35-1866
#35-1867 35-1867 35-1867 #35-1868 35-1868 35-1868 #35-1869 35-1869 35-1869
#35-1870 35-1870 35-1870 #35-1871 35-1871 35-1871 #35-1872 35-1872 35-1872
#35-1875 35-1875 35-1875 #35-1876 35-1876 35-1876 #35-1877 35-1877 35-1877
#35-1877 35-1877 #35-1878 35-1878 35-1878 #35-1878 35-1878 #35-1879 35-1879
35-1879 #35-1879 35-1879 #35-1880 35-1880 35-1880 #35-1880 35-1880 #35-1881

SYMBOL VALUE REFERENCES

		35-1881	35-1881	#35-1881	35-1881	#35-1882	35-1882	35-1882	#35-1882	35-1882
		#35-1887	35-1887	35-1887	#35-1887	35-1887	#35-1888	35-1888	35-1888	#35-1889
		35-1889	35-1889	#35-1890	35-1890	35-1890	#35-1891	35-1891	35-1891	#35-1892
		35-1892	35-1892	#35-1893	35-1893	35-1893	#35-1894	35-1894	35-1894	#35-1896
		35-1896	35-1896	#35-1897	35-1897	35-1897	#35-1898	35-1898	35-1898	#35-1899
		35-1899	35-1899	#35-1900	35-1900	35-1900	#35-1901	35-1901	35-1901	#35-1902
		35-1902	35-1902	#35-1903	35-1903	35-1903	#35-1906	35-1906	35-1906	#35-1907
		35-1907	35-1907	#35-1908	35-1908	35-1908	#35-1909	35-1909	35-1909	#35-1910
		35-1910	35-1910	#35-1911	35-1911	35-1911	#35-1912	35-1912	35-1912	#35-1913
		35-1913	35-1913	#35-1916	35-1916	35-1916	#35-1917	35-1917	35-1917	#35-1918
		35-1918	35-1918	#35-1919	35-1919	35-1919	#35-1919	35-1919	#35-1920	35-1920
		35-1920	#35-1921	35-1921	35-1921	#35-1921	35-1921	#35-1922	35-1922	35-1922
		#35-1922	35-1922	#35-1923	35-1923	35-1923	#35-1923	35-1923	#35-1929	35-1929
		35-1929	#35-1930	35-1930	35-1930	#35-1931	35-1931	35-1931	#35-1932	35-1932
		35-1932	#35-1933	35-1933	35-1933	#35-1934	35-1934	35-1934	#35-1935	35-1935
		35-1935	#35-1936	35-1936	35-1936	#35-1939	35-1939	35-1939	#35-1940	35-1940
		35-1940	#35-1941	35-1941	35-1941	#35-1942	35-1942	35-1942	#35-1943	35-1943
		35-1943	#35-1944	35-1944	35-1944	#35-1945	35-1945	35-1945	#35-1946	35-1946
		35-1946	#35-1949	35-1949	35-1949	#35-1950	35-1950	35-1950	#35-1951	35-1951
		35-1951	#35-1952	35-1952	35-1952	#35-1953	35-1953	35-1953	#35-1954	35-1954
		35-1954	#35-1955	35-1955	35-1955	#35-1956	35-1956	35-1956	#35-1959	35-1959
		35-1959	#35-1960	35-1960	35-1960	#35-1961	35-1961	35-1961	#35-1962	35-1962
		35-1962	#35-1963	35-1963	35-1963	#35-1963	35-1963	#35-1964	35-1964	35-1964
		#35-1965	35-1965	35-1965	#35-1966	35-1966	35-1966			
DCH	= 000004	#8-484	44-2593							
DLE	= 000020	#5-374	38-2140							
DLEF	= 000200	#7-454	37-2094	38-2138	38-2144	41-2325	42-2372	42-2377	42-2395	42-2400
		42-2434	42-2437							
DSC	= 004000	#8-494	42-2439							
EBCACK	000010 R	#11-575	11-592							
EBCCCT	000000 R	#11-567	21-1126							
EBCNAK	000024 R	#11-579	11-593							
EBCRTR	004006 R	11-591	#37-2032							
EBCSYN	000016 R	#11-577	11-588							
EBCXTR	001752 R	11-590	#20-1060							
EBCXT1	001770 R	20-1063	#20-1071							
ECH	= 000121	#6-417	6-418	*16-796	*16-799	*17-842	19-1009	19-1012	*21-1139	
EKT	= 000140	#6-427	6-428	6-429	*22-1189	*26-1475	*39-2184	*41-2346	*41-2360	
ENQ	= 000005	#5-379								
ENQAGN	006220 R	#44-2620								
ENQAG1	006256 R	44-2621	#44-2633							
ENQCNT	= 000010	#4-332	22-1189	26-1475	41-2360					
ENQSET	006300 R	22-1193	44-2634	#44-2639	44-2653					
ENQTIM	006322 R	44-2646	#44-2651							
ENQY	006352 R	44-2652	#44-2657							
ENQZ	006270 R	42-2499	42-2514	#44-2635						
EOT	= 000004	#5-378								
ESNT	= 001000	#8-508	18-924	25-1431						
ETA	003052 RG	3-250	20-1067	#27-1486	37-2034					
ETB	= 000027	#5-382								
ETX	= 000003	#5-377								
ETXF	= 000400	#8-507	16-800	16-806						

SYMBOL	VALUE		REFERENCES							
FBF	= 000400		#8-491	37-2082	41-2314	41-2328				
FRSAT	003736	R	#36-2004	42-2368						
GENQ10	005154	R	41-2340	#41-2352						
GETCHA	004076	R	37-2070	#37-2072						
GETCHR	004064	R	#37-2061	37-2083	37-2104	37-2108	37-2132	38-2153	41-2329	42-2391
			42-2432	42-2480	43-2535					
GETCH1	005410	R	36-2010	42-2370	42-2413	#42-2432	42-2435			
GET1	004106	R	37-2072	#37-2074						
GET1A	004126	R	37-2075	#37-2078						
GET1B	004150	R	37-2079	#37-2082						
GET2	004176	R	37-2086	#37-2088	44-2613					
GET3	004324	R	37-2093	#38-2135						
GET4	004246	R	37-2096	#37-2103						
GET5	004216	R	37-2090	#37-2092						
GET9	004306	R	37-2111	#37-2113						
GNX1	004362	R	38-2139	#38-2144						
GOUOF	004412	R	38-2150	#38-2152						
GTBCC	004232	R	36-1989	36-1997	36-2001	#37-2095	38-2141			
GTBC1	006136	R	44-2590	#44-2597						
GTBC2	006130	R	44-2592	#44-2594						
GTENQ	006040	R	36-1999	#43-2573						
GTENQ1	006056	R	43-2574	#43-2577						
GTEOT	006050	R	37-2081	41-2352	#43-25[illegible]6					
GTETB	005736	R	36-1994	#43-2538						
GTETX	006002	R	36-1995	#43-2560						
GTETO	005744	R	#43-2539	43-2561						
GTET1	005752	R	#43-2540	43-2567	43-2569					
GTFRST	005170	R	37-2091	#42-2365						
GTFRT1	005206	R	36-2004	36-2006	36-2007	36-2008	42-2366	#42-2369		
GTGET	005732	R	#43-2535	43-2557	43-2584	43-2586	44-2595			
GTITB	006012	R	36-1996	#43-2564						
GTITB1	006012	R	#43-2565							
GTIT1	006156	R	44-2598	#44-2603						
GTNONE	004736	R	37-2071	#41-2277						
GTNOPE	005120	R	#41-2339							
GTNO1	004744	R	#41-2280							
GTNO2	004760	R	41-2279	41-2281	#41-2289					
GTNO3	005012	R	41-2292	41-2299	#41-2302					
GTNSAR	004370	R	38-2137	#38-2145						
GTSAT	003712	R	#36-1989	38-2145						
GTSTX	005710	R	36-1990	36-2000	#43-2517					
GTSYN	006070	R	36-1998	#43-2583						
GTUOF	004374	R	37-2106	#38-2149						
HALT	= 000000		#2-4							
INI	= 000400		#7-455	15-737	18-925	23-1220	25-1419	25-1436	25-1440	39-2174
			42-2482	42-2501						
INIENQ	006210	R	42-2502	#44-2616						
INIEN1	006252	R	44-2600	44-2617	#44-2630					
INTBCC	005340	R	42-2396	#42-2401						
INTC1	004676	R	40-2256	40-2258	#40-2260					
INTC2	004716	R	40-2261	40-2263	#40-2265					
INTTIC	004656	R	40-2238	#40-2255						

SYMBOL	VALUE	REFERENCES								
INTVTM	004610 R	15-684	#40-2229	40-2266						
INTVT1	004632 R	#40-2240								
IPL	= 000002	#6-389	6-390							
IRC	= 000062	#6-400	6-401	*37-2058	*37-2084	41-2278				
IRS	= 000036	#5-385								
ISC	= 000072	#6-404	6-405	15-686						
ITB	= 000037	#5-383								
ITBA	= 002000	#7-473	15-722	15-743	16-782	16-791	18-891	19-947	19-963	19-970
		19-977	19-1014	19-1023	24-1378	41-2324	42-2412	43-2539	43-2565	43-2577
		44-2597								
ITS	= 004000	#7-458								
KGBCCR	= 170702	#3-274	18-882	18-895	18-901	19-1018	37-2048	41-2285	43-2555	
KGC	= 000034	#9-539	9-540	17-845	19-945	19-988	37-2052			
KGDATA	= 170704	#3-275	17-816	17-840	19-1009	37-2050	37-2101	41-2306	42-2420	42-2430
		43-2523	43-2533	43-2546						
KGSTAT	= 170700	#3-273	3-274	3-275	17-839	17-845	19-944	19-945	19-987	19-988
		37-2047	37-2049	37-2052	41-2305	41-2307	42-2409	44-2610		
LASBYT	= 000031	#9-536	9-538	9-538						
LCW	= 000006	#6-391	6-392	15-737	*16-779	16-784	*16-786	*16-787	*16-790	18-857
		18-859	18-867	18-874	18-911	18-922	*18-925	18-929	19-956	19-968
		19-999	19-1001	20-1062	*21-1108	21-1109	*21-1125	21-1170	21-1172	*22-1188
		*22-1208	*23-1244	*24-1265	25-1419	25-1421	*25-1436	25-1440	*37-2094	37-2109
		38-2136	38-2138	*38-2144	39-2174	*41-2325	42-2372	*42-2377	42-2395	*42-2397
		*42-2400	42-2414	*42-2434	42-2437	*42-2482	42-2501	42-2512	43-2517	*43-2540
		43-2566	44-2616	44-2620	*44-2622	*44-2633				
LDGRPH	= 000007	#4-364	15-670	42-2498						
LGC	= 000152	#6-433	6-434	6-437	*15-670	*42-2389	*42-2498			
LSI11	= ******	3-242	17-812	17-836	17-844	18-879	18-892	18-898	19-936	19-979
		19-1005	19-1015	24-1303	24-1313	24-1337	37-2043	37-2055	37-2097	41-2282
		41-2302	42-2406	42-2416	42-2426	43-2519	43-2529	43-2542	43-2552	44-2607
LSW	= 000010	#6-392	6-393	*15-715	15-722	*15-741	15-743	16-766	*16-770	*16-778
		16-782	*16-791	16-802	*17-843	*18-891	*18-906	18-913	*18-915	*18-917
		19-947	19-963	19-970	19-977	*19-1014	19-1023	*19-1030	*21-1120	22-1178
		22-1206	24-1258	*24-1264	24-1329	*24-1336	*24-1367	24-1378	24-1391	*24-1393
		25-1412	*25-1418	25-1425	25-1427	25-1438	*26-1474	37-2089	37-2095	37-2107
		38-2149	*38-2157	39-2171	*40-2219	*40-2251	*40-2271	41-2291	41-2320	*41-2324
		41-2339	*42-2393	42-2412	42-2447	42-2510	*43-2539	*43-2565	*43-2568	*43-2577
		43-2583	44-2597	*44-2599						
LTL	= 000776	#6-440	6-441							
MBX	= 000034	#4-336	6-400	15-708	15-711	24-1326				
MCF	= 000004	#7-450	44-2620	44-2622	44-2633					
MDC	= 000050	#9-545	9-546	21-1160	24-1386					
MEND	= 010000	#8-495	43-2560							
MIP	= 100000	#7-478	37-2089	41-2324	42-2393	43-2568	43-2577	44-2599		
MRC	= 000115	#6-414	*15-682	*25-1445	40-2260	*40-2262	*41-2341	*42-2443	*42-2470	
MSRT	000002 R	#4-343	*15-676	25-1445						
MSRTO	004436 R	#39-2170	40-2264							
MSRTOA	004514 R	39-2186	#40-2203							
MSRTOX	004530 R	39-2172	39-2175	#40-2207						
MSRT1	004464 R	#39-2176	42-2492							
MSS	= 000020	#7-452	22-1188	22-1208	23-1240	23-1243	25-1421	25-1436	42-2482	42-2512
		44-2616	44-2622							

SYMBOL	VALUE		REFERENCES						
MTOUT	= 000100		#8-503	39-2170	41-2326	42-2489			
NAK	= 000025		#5-380						
NAKCNT	= 000010		#4-333	16-780	26-1464				
NAKF	= 100000		#8-513	39-2191	42-2457				
NAT	= 001000		#7-456	20-1062	23-1228	23-1231	25-1436	37-2109	42-2482
NEF	= 004000		#7-474	18-915	26-1474	42-2447			
NMP	= 000044		#9-543	9-544	24-1372				
NOP	= 000240		#2-5						
NUB	= 000002		#8-483	38-2151					
OOF	= 000100		#7-453	21-1108	21-1170	23-1248			
OS	= 000000		#3-237	10-552					
OVXM	= 000200		#8-504	44-2623					
PAD	= 000177		#5-384	42-2369					
PARITY	= 010000		#10-562						
PCH	= 000010		#8-485						
PPD	= 000046		#9-544	9-545	19-1026				
PTBSP	000746	R	#18-853	19-995					
PTCON	002326	R	21-1171	#22-1176					
PTC1	002046	R	21-1111	#21-1114					
PTC2	002056	R	21-1115	#21-1118					
PTC3	002346	R	22-1179	#22-1188					
PTC4	002102	R	#21-1122	23-1250					
PTDAB	001730	R	19-1000	#19-1045					
PTDLE	000756	R	#18-857	37-2021					
PTDL1	001006	R	18-858	18-860	#18-863				
PTDZ	001354	R	19-948	#19-952					
PTENQ	001176	R	#18-911	37-2023					
PTENQ1	001226	R	18-914	#18-917					
PTEOT	001244	R	#18-922	37-2024					
PTEXIT	002454	R	22-1204	#22-1213					
PTEXN	002450	R	21-1158	22-1181	22-1207	#22-1211			
PTEXT	002424	R	#22-1202						
PTEX1	002406	R	#22-1195	22-1209					
PTEX3	002406	R	#22-1196	23-1245					
PTITB	001012	R	#18-866	37-2020					
PTIT1	001064	R	18-875	#18-878					
PTIT2	001162	R	#18-904						
PTIT3	001040	R	18-868	18-870	#18-873				
PTIT4	001124	R	18-887	#18-889					
PTIT4A	001120	R	18-885	#18-888					
PTMODE	002264	R	21-1156	#21-1160					
PTMORS	002456	R	21-1112	#23-1217					
PTM1	002602	R	23-1218	#23-1248					
PTM2	002574	R	23-1242	#23-1244					
PTNEXT	000616	R	16-774	#16-805					
PTNORM	000534	R	16-789	#16-793					
PTN0	000600	R	16-798	#16-801					
PTN1	000620	R	16-803	#16-806					
PTR1	000630	R	#17-812	18-871	18-890	19-974	19-976	19-996	
PTR2	000660	R	17-822	#17-827					
PTR3	000634	R	#17-818	18-862	18-877	18-897	18-903	18-931	19-966	19-967	19-1004
PTR4	000676	R	17-829	#17-833					

```
SYMBOL  VALUE              REFERENCES
PTSAT     003762 R          18-854    #37-2016
PTSLVE    002430 R          22-1177   #22-1206
PTSTX     001166 R         #18-906     37-2017
PTSYN     001272 R         #18-929     37-2022
PTS1      002440 R         #22-1208
PTTERM    002620 R          21-1116   #24-1253
PTTEXT    002660 R          24-1259   #24-1264
PTT1      002642 R         #24-1258     24-1261
PUTD0     001310 R          16-801    #19-934
PUTD1     000512 R          16-783     16-785    #16-788
PUTD1A    000422 R         #16-776
PUTD10    001720 R          19-1024   #19-1037
PUTD11    000420 R          16-768    #16-775
PUTD22    001462 R          19-972    #19-975
PUTD23    001430 R          19-964    #19-968
PUTD24    001452 R          19-969    #19-973
PUTD3     001370 R          19-951    #19-955
PUTD4     001516 R          18-872     19-957     19-971     19-978    #19-990     19-998
PUTD5     001540 R          18-863     18-907     18-912     18-916     18-919     18-923     18-926     18-930     19-994
                           #19-996     37-2016    37-2018    37-2019    37-2025
PUTD5A    001544 R          18-932    #19-997
PUTD6     001602 R          17-847     18-888     19-1002   #19-1005
PUTD7     001636 R         #19-1015
PUTD8     001644 R          18-904    #19-1020    19-1054
PUTD9     001710 R          19-1022   #19-1030
RBA     = 000001           #7-464      15-715     22-1206    24-1258    24-1329    24-1336    38-2149
RBC     = 000124           #6-419      6-420      11-594    *42-2402
RBU     = 000004           #6-390      6-391      37-2068
RCVSIB  = 000000           #1-1        4-365      24-1390    24-1399
RCVT    = 100000           #7-461      38-2136    42-2397    42-2414    43-2517    43-2540    43-2566
RCVTMT  = ******  G         42-2507   *42-2508
RCX     = 000020           #6-396      6-397      15-698    *15-707     15-708    *15-710
RDBF    = 001000           #7-472      25-1412    25-1418    40-2219
REPEAT    005146 R         #41-2349    42-2456
RETNX   =%000007           #3-270
RIB     = 000137           #6-426
RJT     = 000040           #7-469      16-770     16-802     17-843
RKT     = 000141           #6-428     *16-780     26-1461   *42-2453
ROVR    = 000100           #8-488      37-2077
RSTS    = ******            3-227      3-251      14-628     15-717     19-936     19-958     19-979     19-1031    19-1036
                            21-1091    22-1180    22-1184    24-1257    24-1332    26-1457    37-2042    37-2055    37-2110
                            37-2116    39-2176    39-2192    41-2277    41-2298    41-2310    41-2342    41-2356    42-2374
                            42-2403    42-2422    42-2442    42-2449    42-2458    42-2503    43-2525    43-2548    44-2624
                            45-2659
RSW     = 000012           #6-393      6-394      15-728     15-729    *15-732    *15-745    *24-1266   *37-2073   *37-2077
                            37-2078   *37-2080    37-2082    37-2103   *38-2151   *38-2152   *40-2272   *41-2314   *41-2328
                           *42-2398   *42-2439   *42-2483   *43-2538   *43-2560    43-2573   *43-2576   *44-2593
RSX     = 000001           #1-3        3-235      3-251      10-549     12-607     24-1273    24-1346    27-1479    42-2442
RSX11M  = ******            14-639
RT11    = ******            3-231      3-251
RVI     = 040000           #8-512      42-2474
R3780   = ******            9-532      11-583     36-1980    36-1983    36-1991    37-2026
```

SYMBOL	VALUE		REFERENCES								
R6	=%000006		#3-264								
R7	=%000007		#3-265								
SBTCCT	= 000000		#12-606								
SCC	= 000277		#2-15								
SEC	= 000261		#2-11								
SEN	= 000270		#2-14	3-279							
SENV	= 000272		#3-279								
SETACK	005524	R	42-2379	#42-2468							
SETAC1	005532	R	#42-2469	42-2475							
SETARF	005262	R	42-2373	#42-2389							
SETDLE	005414	R	36-2009	38-2142	#42-2434						
SETENQ	005652	R	36-2011	#42-2501							
SETEOF	005554	R	42-2438	#42-2477							
SETEOT	005424	R	36-2012	#42-2437							
SETETB	002176	R	21-1135	21-1137	#21-1139						
SETJ	005706	R	42-2511	#42-2514							
SETMIP	005274	R	36-2005	#42-2393							
SETM1	005404	R	42-2415	#42-2426							
SETNAK	005446	R	36-2013	#42-2442	42-2490						
SETN1	005470	R	#42-2455	42-2513							
SETRVI	005544	R	42-2385	#42-2474							
SETSYN	002136	R	#21-1131								
SETWAK	005160	R	#41-2356	42-2383							
SETWRG	005622	R	42-2381	#42-2489							
SEV	= 000262		#2-12	3-279							
SEZ	= 000264		#2-13								
SIS	= 000010		#7-451								
SOH	= 000001		#5-376								
SRC	= 000153		#6-434	*38-2156	*40-2247	40-2255	*40-2257				
SRTA	= 000400		#7-471	38-2157	40-2251	40-2271	41-2320	43-2583			
SRTOF	= 000200		#8-489	37-2078	37-2080	40-2272					
SSRT	000004	R	#4-350	*15-677	40-2247						
SSRTM	004640	R	#40-2245	42-2394	43-2585						
SSRTO	004720	R	40-2259	#40-2271							
STARF1	005640	R	42-2390	#42-2495							
STX	= 000002		#5-375								
STXBCC	005726	R	43-2518	#43-2529							
STXF	= 000020		#7-468	18-906	18-913	26-1474					
SYN	= 000026		#5-381								
SYP	= 000032		#9-538	9-539	21-1140						
TBC	= 000070		#6-403	6-404	*16-794	*18-869	*19-997				
TBS	= 000112		#6-412	6-413	*15-690	19-953					
TCI3	002112	R	#21-1125								
TCJ1	002506	R	21-1173	#23-1226							
TCJ2	002532	R	23-1227	23-1230	#23-1233						
TCJ3	002556	R	23-1234	23-1237	#23-1240						
TCP	= 000130		#6-422	6-423	*15-687	22-1191	24-1364	44-2655			
TCT	= 000120		#6-416	6-417	*15-678	24-1395	41-2295				
TCW	= 000142		#6-429	6-430	*21-1118	*21-1122	22-1176	22-1199	*23-1222	*23-1223	23-1241
			*23-1249								
TDENT	000532	R	19-1038	25-1423	#26-1461	39-2194	42-2461	44-2626			
THALT	= 000240		3-281	41-2300							

SYMBOL	VALUE	REFERENCES							
THC	= 000116	#6-415	6-416	17-833	*18-878				
TIB	= 000154	#6-437	6-440	15-689					
TICINT	000006 R	#4-357	*15-675	40-2234					
TRANER	005476 R	41-2347	42-2448	42-2454	#42-2457				
TRAN1	005504 R	#42-2458	42-2472						
TRAN2	005514 R	#42-2465	42-2493						
TRC	= 000114	#6-413	6-414	6-415	17-828	*18-873	18-884	*19-1029	
TRNREC	= 001000	#8-492	42-2398						
TRYNAK	000174 R	24-1354	#24-1372						
TSA	= 000066	#6-402	6-403	*16-793	19-955	26-1468			
TSW	= 000014	#6-394	6-395	*16-800	*16-806	*18-918	*18-924	*19-935	19-1021
		25-1431	25-1449	*25-1451	26-1467	*26-1476	*39-2170	*39-2191	*41-2326
		*42-2468	*42-2474	42-2478	42-2489	*44-2623	44-2651		*24-1267 *42-2457
TTM	= 000110	#6-411	6-412						
TT2A	= 000144	#6-430	6-431	25-1415	*40-2211				
TT2C	= 000146	#6-431	6-432	25-1416	*40-2212				
TT3A	= 000150	#6-432	6-433	25-1417	*40-2213				
TXC	= 000064	#6-401	6-402	*16-781	*17-820				
T1A	= 000074	#6-405	6-406	*21-1140					
T1C	= 000076	#6-406	6-407	*21-1141					
T2A	= 000100	#6-407	6-408	19-949	*19-954	*24-1355	*24-1358	*24-1372	*25-1415 40-2211
		*40-2221	*40-2222						
T2C	= 000102	#6-408	6-409	*17-819	17-821	*17-833	18-878	18-886	19-950 *19-952
		*24-1359	*24-1373	*25-1416	40-2212	*40-2223			
T3A	= 000104	#6-409	6-410	*19-1011	*19-1012	*19-1047	*19-1048	*24-1361	*24-1362 *25-1417
		40-2213	*40-2224						
T3C	= 000106	#6-410	6-411	*19-1013	*19-1049	*24-1363			
UAPTIM	= 000000	#3-236	4-323	4-342	4-349	4-352	4-356	15-671	25-1444 40-2233
		40-2246	44-2641						
UBL	= 000026	#6-399	6-400	15-700	24-1302				
UOVR	= 000020	#8-486	15-732	37-2103	38-2152				
XAC	= 010000	#7-475	16-766	19-1030	22-1178	26-1474	41-2324		
XMTTMT	= ****** G	39-2181	*39-2182						
XPT	= 020000	#7-460	16-784	16-787	23-1235	23-1238			
XRC	= 000001	#7-448	7-449	21-1123					
XXP	= 010000	#7-459	16-786	18-857	18-867	18-874	18-929	19-968	19-1001
X.	= 000002	#7-449							
X1	001776 R	20-1068	#20-1073						

BISYNC CREATED BY MACRO ON 6-DEC-77 AT 08:18 PAGE 12
MACRO CROSS REFERENCE CREF

MACRO NAME REFERENCES

HEX

#35-1786 #35-1804 #35-1805 #35-1806 #35-1807 #35-1808 #35-1809 #35-1810 #35-1811 #35-1814
#35-1815 #35-1816 #35-1817 #35-1818 #35-1819 #35-1820 #35-1821 #35-1824 #35-1825 #35-1826
#35-1827 #35-1829 #35-1830 #35-1831 #35-1832 #35-1835 #35-1836 #35-1837 #35-1838 #35-1839
#35-1840 #35-1841 #35-1842 #35-1845 #35-1846 #35-1847 #35-1848 #35-1849 #35-1850 #35-1851
#35-1852 #35-1855 #35-1856 #35-1857 #35-1858 #35-1859 #35-1860 #35-1861 #35-1862 #35-1865
#35-1866 #35-1867 #35-1868 #35-1869 #35-1870 #35-1871 #35-1872 #35-1875 #35-1876 #35-1877
#35-1878 #35-1879 #35-1880 #35-1881 #35-1882 #35-1887 #35-1888 #35-1889 #35-1890 #35-1891
#35-1892 #35-1893 #35-1894 #35-1896 #35-1897 #35-1898 #35-1899 #35-1900 #35-1901 #35-1902
#35-1903 #35-1906 #35-1907 #35-1908 #35-1909 #35-1910 #35-1911 #35-1912 #35-1913 #35-1916
#35-1917 #35-1918 #35-1919 #35-1920 #35-1921 #35-1922 #35-1923 #35-1929 #35-1930 #35-1931
#35-1932 #35-1933 #35-1934 #35-1935 #35-1936 #35-1939 #35-1940 #35-1941 #35-1942 #35-1943
#35-1944 #35-1945 #35-1946 #35-1949 #35-1950 #35-1951 #35-1952 #35-1953 #35-1954 #35-1955
#35-1956 #35-1959 #35-1960 #35-1961 #35-1962 #35-1963 #35-1964 #35-1965 #35-1966

CLOSE MACRO M1104 06-DEC-77 08:19 PAGE 1

```
                        ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                  RSX=1
000001                  RSX11M=1
000002                  NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                  NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                        ;
                        ;MACRO DEFINITIONS
000000                          TAGNUM=0
                                .MACRO  ERROR   TEXT
                                ERRORK  <TEXT>
                                JSR     PC,PROMPT
                                .ENDM
                        ;
                        ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                        ;
                                .MACRO  ERRORK  TEXT
                                TAGNUM=TAGNUM+1
                                .IRP    N,<\ TAGNUM>
                                JSR     R5,TYPMSG
                                .WORD   TAG'N
                                .WORD   TAGE'N-TAG'N
                                .PSECT  MESAG2,GBL
                        TAG'N:  .ASCII  "TEXT"
                        TAGE'N:
                                .PSECT  MAIN
                                .ENDM
                                .ENDM

                        ; MACRO TO QUEUE A MESSAGE ONLY
                        ;
                                .MACRO  ERRORM  TEXT
                                TAGNUM=TAGNUM+1
                                .IRP    N,<\ TAGNUM>
                                JSR     R5,TYPMS1
                                .WORD   TAG'N
                                .WORD   TAGE'N-TAG'N
                                .PSECT  MESAG2,GBL
                        TAG'N:  .ASCII  "TEXT"
                        TAGE'N:
                                .PSECT  MAIN
                                .ENDM
                                .ENDM

                        ; MACRO TO QUEUE A MESSAGE AND PROMPT
                        ;
                                .MACRO  ERRORP  TEXT
                                ERRORM  <TEXT>
                                JSR     PC,PROMPT
                                .ENDM

                        ;
                                .MACRO  FATAL   TEXT
                                ERRORK  <TEXT>
                                JSR     R5,DIE
                                .ENDM
```

```
                                .TITLE  CLOSE
                                .IDENT  /00006/
                        ;
                        ; COPYRIGHT (C)  1975,1976,1977
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                        ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                        ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                        ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                        ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                        ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                        ;
                        ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ; DEC.
                        ;
                        ; VERSION 0
                        ;
                        ; ROBERT H.  ROSENBAUM 12-AUGUST-74
                        ;
                        ; LAST MODIFICATION ON 29-MAY-75
                        ;
                        ; ROUTINES TO CLOSE RECEIVE FILES
                        ;
                        ; MACRO CALLS
                                .MCALL  CLOSE$,FCSBT$,FDOFF$,CALL,RETURN
000000                          FCSBT$  DEF$L
000000                          FDOFF$  DEF$L
                        ;
                        ; INPUTS:
                        ;       NONE
                        ; OUTPUTS:
                        ;       BOTH INPUT FILES ARE CLOSED AND THE RECEIVE BLOCK AREA
                        ;       IS UPDATE TO POINT TO THE NEXT FILE TO BE WRITTEN
                        ;
000000                          .PSECT  MAIN
                        ;
000000                  CLOSER::
                                .IF DF  RT11
                                MOV     #CHANL,R5
                                JSR     PC,FILBUF
                                .WAIT   12
                                .CLOSE  12
                                .ENDC
                                .IF DF  RSX
000000  012700  000000G        MOV     #FDBOU0,R0      ;GET THE FIRST FDB ADDRESS
000004                          CALL    DLTZER          ;DELETE THE FILE IF IT IS EMPTY
000010  103416                 BCS     20$             ;IT WAS, NOW CHECK FOR SELECTIVE RECV
000012                          CLOSE$                  ;NOT EMPTY - CLOSE THE FILE
                                .ENDC
```

```
                                                TSTB    RCHN            ;RECEIVE CHAIN MODE?
000016  105767  000000G                       BNE     20$             ;YES - DON'T INCREMENT TO NEXT FILE
000022  001011                                .IF DF  RT11
                                              ADD     #12,RCVBLK      ;NO, GET NEXT RECEIVE FILE
                                              .ENDC
                                              .IF DF  RSX
000024  005777  000000G                       TST     @RCVBLK         ;ANY MORE RECEIVE FILES?
000030  100406                                BMI     20$             ;NO, - DON'T INCREMENT TO NEXT FILE
                                                                      ; SINCE THERE IS NONE!
000032  105767  000000G                       TSTB    INCRCV          ;INCOMPLETE RECEIVE PENDING?
000036  100403                                BMI     20$             ;YES, DON'T INCREMENT TO NEXT FILE
000040  062767  000036G 000000G               ADD     #S.FNB+RCVSWT,RCVBLK    ;NO, GET NEXT RECEIVE FILE
                                              .ENDC
                                              .IF DF  RT11
                                      20$:    TSTB    SWRSL           ;SELECTIVE RECEIVE?
                                              BEQ     30$             ;NO
                                              MOV     #CHANL2,R5
                                              JSR     PC,FILBUF
                                              .ENDC
                                              .IF DF  RT11
                                              .WAIT   13
                                              .CLOSE  12
                                              .ENDC
                                              .IF DF  RSX
000046  105767  000000G              20$:    TSTB    SWRSL           ;SELECTIVE RECEIVE ACTIVE?
000052  001411                                BEQ     30$             ;NO
000054  012700  000000G                       MOV     #FDBOU1,R0      ;PUT PRINT FILE'S FDB IN R0
000060                                        CALL    DLTZER          ;YES, DELETE THE PRINT FILE IF EMPTY
000064  103402                                BCS     25$             ;IT WAS
000066                                        CLOSE$                  ;IN CASE SELECTIVE RECEIVE ACTIVE
000072  005060  000120              25$:    CLR     F.FVER(R0)      ;CLEAR THE VERSION # SO FCS
                                                                      ; WILL USE THE NEXT AVAILABLE
                                              .ENDC
000076  000207                      30$:    RTS     PC              ;RETURN
```

```
                                        ;+
                                        ; **-DLTZER-DELETE A FILE IF IT IS ZERO LENGTH
                                        ;
                                        ; INPUTS:
                                        ;       R0=FDB ADDRESS
                                        ;-
000100  132760  000001  000017  DLTZER: BITB    #FD.REC,F.RCTL(R0) ;IS THIS A RECORD-ORIENTED DEVICE?
000106  001016                          BNE     10$                ;YES, JUST CLOSE THE FILE
000110  005760  000010                  TST     F.EFBK(R0)         ;LAST BLOCK NUMBER 1?
000114  001013                          BNE     10$                ;NO
000116  022760  000001  000012          CMP     #1,F.EFBK+2(R0)    ; ..
000124  001007                          BNE     10$                ; ..
000126  005760  000014                  TST     F.FFBY(R0)         ;FIRST FREE BYTE 0?
000132  001004                          BNE     10$                ;NO
000134                                  CALL    .DLFNB             ;ALL ZERO - FILE IS EMPTY
                                                                   ; SO DELETE IT AND USE IT AGAIN FOR
                                                                   ; NEXT RECEIVE OPERATION
000140  000261                          SEC                        ;INDICATE TO CALLER THAT FILE
                                                                   ; WAS DELETED
000142  000401                          BR      20$                ;RETURN
000144  000241                  10$:    CLC                        ;INDICATE THAT FILE WAS NOT EMPTY
000146                           20$:    RETURN
```

```
                                ; ROUTINE TO CLOSE TRANSMIT FILE
                                ;
                                ;+
                                ; INPUTS:
                                ;       NONE
                                ; OUTPUTS:
                                ;       INPUT FILE IS CLOSED
                                ;-
000150                          CLOSET::
                                        .IF DF  RSX
000150                                  CLOSE$  #FDBIN          ;CLOSE THE FILE
                                        .ENDC
000160  000207                          RTS     PC              ; AND RETURN
                                ;
000001                                  .END
```

```
CLOSE   MACRO M1104  06-DEC-77 08:19  PAGE 4-1
SYMBOL TABLE

CH.AND= 000001          FD.ISP= 002000          F.BKDS= 000020          F.RATT= 000001          N.FNAM= 000006
CLOSER  000000RG    002 FD.MNT= 100000          F.BKEF= 000050          F.RCNM= 000034          N.FTYP= 000014
CLOSET  000150RG    002 FD.OSP= 004000          F.BKP1= 000051          F.RCTL= 000017          N.FVER= 000016
DLTZER  000100R     002 FD.PLC= 000004          F.BKST= 000024          F.RSIZ= 000002          N.NEXT= 000022
FA.APD= 000100          FD.PSE= 010000          F.BKVB= 000064          F.RTYP= 000000          N.STAT= 000020
FA.CRE= 000010          FD.RAH= 000001          F.CHR = 000075          F.SEQN= 000100          N.UNIT= 000034
FA.DLK= 001000          FD.RAN= 000002          F.CNTG= 000034          F.SPDV= 000072          PAR$$$= 000027
FA.ENB= 100000          FD.REC= 000001          F.DFNB= 000046          F.SPUN= 000074          RCHN  = ****** GX
FA.EXC= 002000          FD.RWM= 000001          F.DSPT= 000044          F.STBK= 000036          RCVBLK= ****** GX
FA.EXT= 000004          FD.SDI= 000020          F.DVNM= 000134          F.UNIT= 000136          RCVSWT= ****** GX
FA.NSP= 000100          FD.SQD= 000040          F.EFBK= 000010          F.URBD= 000020          RSX   = 000001
FA.POS= 010000          FD.TTY= 000004          F.EFN = 000050          F.VBN = 000064          RSX11M= 000001
FA.RD = 000001          FD.WBH= 000002          F.EOBB= 000032          F.VBSZ= 000060          R.FIX = 000001
FA.RWD= 004000          FF.CHR= 000005          F.ERR = 000052          INCRCV= ****** GX       R.SEQ = 000003
FA.SEQ= 040000          FF.NV = 000003          F.FACC= 000043          NB.DEV= 000200          R.VAR = 000002
FA.SHR= 000040          FF.POE= 000002          F.FFBY= 000014          NB.DIR= 000100          SWRSL = ****** GX
FA.TMP= 000020          FF.RWD= 000001          F.FNAM= 000110          NB.NAM= 000004          S.FATT= 000016
FA.WCK= 020000          FF.RWF= 000006          F.FNB = 000102          NB.SD1= 000400          S.FDB = 000140
FA.WRT= 000002          FF.SPC= 000004          F.FTYP= 000116          NB.SD2= 001000          S.FNAM= 000006
FDBIN = ****** GX       FO.APD= 000106          F.FVER= 000120          NB.SNM= 000040          S.FNB = 000036
FDBOU0= ****** GX       FO.MFY= 000002          F.HIBK= 000004          NB.STP= 000020          S.FNBW= 000017
FDBOU1= ****** GX       FO.RD = 000001          F.LUN = 000042          NB.SVR= 000010          S.FNTY= 000004
FD.BLK= 000010          FO.UPD= 000006          F.MBCT= 000054          NB.TYP= 000002          S.FTYP= 000002
FD.CCL= 000002          FO.WRT= 000016          F.MBC1= 000055          NB.VER= 000001          S.NFEN= 000020
FD.COM= 020000          F.ACTL= 000076          F.MBFG= 000056          NRCVSF= 000000          TAGNUM= 000000
FD.CR = 000002          F.ALOC= 000040          F.NRBD= 000024          NTRNSF= 000002          .CLOSE= ****** G
FD.DIR= 000010          F.BBFS= 000062          F.NREC= 000030          N.DID = 000024          .DLFNB= ****** GX
FD.FTN= 000001          F.BDB = 000070          F.OVBS= 000030          N.DVNM= 000032          ...GBL= 000000
FD.F11= 040000          F.BGBC= 000057          F.RACC= 000016          N.FID = 000000          ...TPC= 000140
FD.INS= 000010          F.BKDN= 000026

. ABS.  000000      000
        000000      001
MAIN    000162      002
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  2523 WORDS  ( 10 PAGES)
DYNAMIC MEMORY:  2810 WORDS  ( 10 PAGES)
ELAPSED TIME:  00:00:16
 256,020 CLOSE;27, 256,030 CLOSE/CRF/-SP=UAPRXM,CLOSE
```

CLOSE CREATED BY MACRO ON 6-DEC-77 AT 08:19 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES					
CLOSER	000000	RG	#2-45					
CLOSET	000150	RG	#4-124					
DLTZER	000100	R	2-54	2-85	#3-99			
FDBIN	= ******	GX	4-126	4-126				
FDBOU0	= ******	GX	2-53					
FDBOU1	= ******	GX	2-84					
FD.REC	= 000001		3-99					
F.EFBK	= 000010		3-101	3-103				
F.FFBY	= 000014		3-105					
F.FVER	= 000120		*2-88					
F.RCTL	= 000017		3-99					
INCRCV	= ******	GX	2-67					
NRCVSF	= 000000		#1-5					
NTRNSF	= 000002		#1-4					
PAR$$$	= 000027		#4-126	4-126				
RCHN	= ******	GX	2-58					
RCVBLK	= ******	GX	2-64	*2-69				
RCVSWT	= ******	GX	2-69					
RSX	= 000001		#1-2	2-52	2-63	2-81	4-125	
RSX11M	= 000001		#1-3					
RT11	= ******		2-46	2-60	2-71	2-77		
SWRSL	= ******	GX	2-82					
S.FNB	= 000036		2-69					
TAGNUM	= 000000		#1-8					
.CLOSE	= ******	G	2-56	2-56	2-87	2-87	4-126	4-126
.DLFNB	= ******	GX	3-107					

MACRO NAME	REFERENCES					
CALL	#2-33	#2-54	#2-85	#3-107		
CLOSE$	#2-33	#2-56	#2-87	#4-126		
ERROR	#1-9					
ERRORK	#1-16					
ERRORM	#1-32					
ERRORP	#1-47					
ERR$	#2-56	#2-56	#2-87	#2-87	#4-126	#4-126
FATAL	#1-53					
FCSBT$	#2-33	#2-34				
FDOFF$	#2-33	2-35				
LOFDB$	#2-56	2-56	#2-87	2-87	#4-126	4-126
RETURN	#2-33	#3-114				

DELETE MACRO M1104 06-DEC-77 08:20 PAGE 1

```
                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM


                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                                .TITLE  DELETE
                                .IDENT  /00001/
                        ;
                        ; COPYRIGHT (C)  1975, 1976, 1977
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                        ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                        ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                        ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                        ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                        ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                        ;
                        ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ; DEC.
                        ;
                        ; VERSION 0
                        ;
                        ; ROBERT H.  ROSENBAUM 12-AUGUST-74
                        ;
                        ; DELETE OUTPUT FILES
                        ;
                        ; MACRO CALLS
                                .MCALL  CALL
                        ;
000000                          .PSECT  MAIN
                        ;+
                        ; INPUTS:
                        ;       NONE
                        ; OUTPUTS:
                        ;       FILES WITH FDB'S AT LABELS FDBOU0 & FDBOU1 ARE DELETED
                        ;       AND CLOSED
                        ;-
000000                  DELETE::
000000  012700  000006G          MOV     #FDBOU0,R0      ;PUT ADDR OF FDB INTO R0
000004                          CALL    .DLFNB          ; AND DELETE FIRST FILE BY FNB
000010  012700  000006G          MOV     #FDBOU1,R0      ;PUT ADDR OF FDB INTO R0
000014                          CALL    .DLFNB          ; AND DELETE SECOND FILE BY FNB
000020  000207                  RTS     PC              ;AND RETURN
                        ;
        000001                  .END
```

```
DELETE  MACRO M1104  06-DEC-77 08:20  PAGE 2-1
SYMBOL TABLE

DELETE  000000RG    002 FDBOU1= ****** GX       NTRNSF= 000002          RSX11M= 000001          .DLFNB= ****** GX
FDBOU0= ****** GX       NRCVSF= 000000          RSX   = 000001          TAGNUM= 000000

. ABS.  000000      000
        000000      001
MAIN    000022      002
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  416 WORDS  ( 2 PAGES)
DYNAMIC MEMORY:  2810 WORDS  ( 10 PAGES)
ELAPSED TIME:  00:00:04
 256,020 DELETE;27, 256,030 DELETE/CRF/-SP=UAPRXM,DELETE
```

```
DELETE     CREATED BY  MACRO  ON 6-DEC-77 AT 08:20      PAGE 1
SYMBOL CROSS REFERENCE                                  CREF
SYMBOL  VALUE              REFERENCES
DELETE    000000 RG        #2-41
FDBOU0  = ******  GX         2-42
FDBOU1  = ******  GX         2-44
NRCVSF  = 000000             #1-5
NTRNSF  = 000002             #1-4
RSX     = 000001             #1-2
RSX11M  = 000001             #1-3
TAGNUM  = 000000             #1-8
 DLFNB  = ******  GX         2-43        2-45
```

DELETE CREATED BY MACRO ON 6-DEC-77 AT 08:20 PAGE 2
MACRO CROSS REFERENCE CREF

MACRO NAME	REFERENCES		
CALL	#2-31	2-43	2-45
ERROR	#1-9		
ERRORK	#1-16		
ERRORM	#1-32		
ERRORP	#1-47		
FATAL	#1-53		

DPV112 MACRO M1104 06-DEC-77 08:20 PAGE 1

```
                                        ;PREFIX FILE FOR RSX-11M DP11 OR DU11 ISR
000001                                  RSX=    1       ;SET FOR CUTBACK CODE JAN '77
000001                                  RSX11M=1
000000                                  RCVSIB=0
000000                                  CRBINT=0
```

```
                                ;PREFIX FILE FOR RSX11M ASSEMBLER AND ANY OTHER ASSEMBLER
                                ;WHICH DOES NOT RECOGNIZE OPCODE MNEMONICS EXCEPT WHEN THEY
                                ;APPEAR IN THE OPCODE FIELD
000000                          HALT=0
000240                          NOP=240
000241                          CLC=241
000242                          CLV=242
000244                          CLZ=244
000250                          CLN=250
000257                          CCC=257
000261                          SEC=261
000262                          SEV=262
000264                          SEZ=264
000270                          SEN=270
000277                          SCC=277
```

```
                        ;COMTEX-11 DP11 SYNCHRONOUS LINE ADAPTER ISR
                        ;
                        ; COPYRIGHT (C)  1975, 1976, 1977
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS  01754
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                        ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                        ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                        ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                        ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                        ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                        ;
                        ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ; DEC.
                        ;
                        ;V01-01 1/9/74 HMS
                        ;SUPPORT ADDED FOR RSTS, RSX11D
                        ;V01-02 7/2/74 RHR
                        ;SUPPORT ADDED FOR RSX11M
                        ;V01-03 7/23/74 JLH
                        ;CORRECTIONS TO CP.RST, CP.KLL
                        ;DISCONNECT FROM VECTOR IN CP.RST
                        ;V01-04 7/24/74 RHR
                        ;MOVE CONNECT TO INTERRUPT CODE FOR RSX11M SYSTEM INTO
                        ;THE EXEC'S BUFFER POOL.  AS A RESULT THIS CODE FOR RSX11D
                        ;AND RSX11M LOOKS THE SAME.  THE MAIN DIFFERENCE IS THAT
                        ;..CINT IS A ROUTINE IN THE EXEC FOR RSX11D WHERE AS IT
                        ;IS A ROUTINE IN THE USERS AREA (DEFINED IN A SEPARATE MODULE)
                        ;FOR RSX11M
                        ;V01-05 10/15/74 RHR
                        ;REPLACE OFFSET (60000) IN TABLE FOR ADDRESSES OF CRB AND SIB
                        ;LOADERS WHEN ASSEMBLED FOR RSX-11D
                        ;V01-06 1/22/75 JLH
                        ;CORRECT DEFINITIONS OF DEVICE ADDRESS AND VECTOR FOR RSTS/E
                        ;V01-07 7/2/75 RHR
                        ;MAKE DEVICE PRIORITY A GLOBAL DEFINITION FOR RSX
                        ;V01-10 9/5/75 BLR
                        ;MODIFY FOR RT11 F/B
                        ;V01-11 7/21/76 CEP
                        ;MODIFY TO RUN WITH V6B OF RSTS/E
                        ;V01-12 8/77 CEP
                        ;MODIFY FOR CUT-CODE VERSION
                        ;
                                .TITLE DPV112

                        ;TABLE PORTION , INCLUDES DEFINTIONS, ISR TABLES, VECTOR DEFINITIONS

                                .IF DF RSX11D!RSX11M!RT11
```

```
000000                  OS=0
                                .ENDC
                                .IF DF RSTS
                        OS=0
                                .ENDC


                        ;DP11 ISR GLOBALS

                                .GLOBL C.SPS,C.SARR
                                .GLOBL CPR2,CPT2
                                .GLOBL CP.OSU,CP.MDM,CP.MOD,CP.INI,CP.KLL
                                .IF DF  RT11
                                .MCALL  .DEVICE,..V2..
                                ..V2..
                                .GLOBL  ..CINT,..DINT,CFKLU,RMON
                                .ENDC
                                .IF DF RSTS
                                .GLOBL  CFKLU,CSR.RJ
                                .ENDC

                                .IF DF OS
000000                  LNO=    0               ;DP11 LINE NUMBER =- FOR RSTS/RSX11D
                                .ENDC
000001                  CPD26=1                 ;DEFINE AS 1 TO CONFIGURE DP11 ISR
                                                ;FOR DECCOMM11D26 2780 EMULATOR
                                                ;SYSTEM, (LINE 2, DP11 RSR ADDRESS
                                                ;=174770, VECTORS DEFINED FROM
                                                ;300 THRU 400).

                                                ;DEFINE AS 0 TO CONFIGURE FOR
                                                ;GENERAL DP11 SUPPORT.  LINE 1:
                                                ;DP11 RSR=174760, VECTOR 310, GENERAL
                                                ;CRB ROUTINES (CRBX).
                                                ;LINE 2:  DP11 RSR=174770, VECTOR 300,
                                                ;CRB2 ROUTINES.
                                                ;ADDITIONAL DP11'S CAN BE SUPPORTED
                                                ;BY ADDING ADDITIONAL VECTOR DEFINITIONS,
                                                ;ISR LINE TABLES AND CRB ROUTINES.


                        ;START OF DP11 INTERRUPT VECTOR PAIRS
                        ;**********
                        ;DEFINITIONS OF REGISTERS, ETC.
                                PS=177776
177776                  000000          R0=%0
000000                  000001          R1=%1
000001                  000002          R2=%2
000002                  000003          R3=%3
000003                  000004          R4=%4
000004                  000005          R5=%5
000005                  000006          R6=%6
000006                  000007          R7=%7
000006                  000006          SP=%6
000007                  000007          PC=%7
```

```
115          000240                          NOP=240
116          000254                          CLZN=254
117          000274                          SEZN=274
118          000252                          CLVN=252
119
120
```

```
                                ;DP11 ISR DEFINITIONS

                                        .IF DF  RSX11D!RSX11M!RT11
                                        .GLOBL  CPRSR
                                        .GLOBL  CPVP            ;VECTOR ADDRESS GLOBALLY DEFINED
                                        .GLOBL  DE.PRI          ;DEVICE PRIORITY GLOBALLY DEFINED
                                        .ENDC
                                        .IF DF RSTS
                                        DE.PRI=300              ;DEFAULT TO LEVEL 6
                                        .ENDC
000000                                  CPRTC=0                 ;TYPE CODE FOR INPUT FROM DP11

010000                                  CPB12=10000
020000                                  CPB13=20000
040000                                  CPB14=40000
100000                                  CPB15=100000
                                        .IF DF RSX11D
                                        .PSECT ISR,CON
                                        OFF=60000
                                        .IFF
000000                                  OFF=0
                                        .ENDC
                                        .IF DF RSX11M
000000                                  .PSECT ISR,CON
                                        .ENDC
```

```
                                ; INDEX DEFINTIONS FOR DP11 LINE TABLE
                                ;
                                ; THESE GIVE THE DISPLACMENTS OF THE VARIOUS PARAMETERS
                                ; FROM THE START OF THE ISR LINE PARAMETER AREA FOR EACH LINE
                                ; WHEN USED BY THE INTERRUPT ROUTINES THEY REQUIRE AN OFFSET
                                ; THE OFFSETS ARE R FOR RECEIVE AND X FOR TRANSMIT
                                ; THESE ARE USED IN THE FORM AAA-R OR AAA-X IN THE INTERRUPT ROUTINES
                                ; THUS PRESERVING A CONSISTENT MNEMONIC USAGE FOR EACH INDEX WHILE
                                ; PROVIDING THE CORRECTLY OFFSET VALUE REQUIRED BY THE IMPLEMENTATION
000004                          R=4             ;SUBTRACTED FROM ANY INDEX USED BY CPRIS
000010                          X=10            ;SUBTRACTED FROM ANY INDEX USED BY CPTIS

000010                          SRA=X                   ;DP11 STATUS REGISTER ADDRESS
000012                          LNR=SRA+2               ;(BYTE) LINE NUMBER
000013                          STC=LNR+1               ;(BYTE) SIB TYPE CODE
000014                          XPR=STC+1               ;INTERRUPT SERVICE PRIORITY - XMIT
000016                          XBR=XPR+2               ;BR BUSY WORD IN SCHEDULER  - XMIT
000020                          RPR=XBR+2               ;INTERRUPT SERVICE PRIORITY - RCV
000022                          RBR=RPR+2               ;BR BUSY WORD IN SCHEDULER  - RCV
000024                          CBL=RBR+2               ;CRB LOADER FOR THIS LINE
000026                          CBU=CBL+2               ;CRB UNLOADER FOR THIS LINE
000030                          CBI=CBU+2               ;CRB INIT ROUTINE FOR THIS LINE
000032                          SBL=CBI+2               ;SIB LOADER FOR THIS LINE
000034                          XWP=SBL+2               ;POINTER TO NEXT XMIT WORD PAIR
000036                          XBA=XWP+2               ;CURRENT XMIT BUFFER ADDRESS
000040                          XBC=XBA+2               ;CURRENT XMIT RESIDUAL BYTE COUNT
000042                          SSC=XBC+2               ;STRIP SYNCH CONTROL FLAG
000044                          WBF=SSC+2               ;WORD/BYTE SELECT FLAG
000046                          CPTL=WBF+2              ;SIZE OF ISR LINE TABLE M BYTES.
```

```
                                        ;START OF DP11 INTERRUPT SERVICE
                                                .IF NDF RSTS
000000'                                 CPAS=.
                                        ;THIS SERIES OF CODING WOULD BE CONTINUED FOR
                                        ;ALL DP11 LINES IN THE SYSTEM.
                                        ;**********
                                        ;**********
                                        ;LINE TABLE FOR DP11 #2
                                        ;
                                        ;

000000                                  CPR2:
                                                .IF DF  RT11
                                                .GLOBL  DC.PRI
                                                JSR     R5,@RMON        ;NOTIFY RT-11 OF INTERRUPT & CHNG PRIOR.
                                                                        ;ALSO SAVES R5 & R4
                                                .WORD   DC.PRI          ;ARGUMENT FOR INTEN
                                                .ENDC
000000  004267  000646                          JSR     R2,CPRIS
000004                                  CPT2:
                                                .IF DF  RT11
                                                JSR     R5,@RMON        ;NOTIFY RT-11 OF INTERRUPT & CHNG PRIOR.
                                                                        ;ALSO SAVES R5 & R4
                                                .WORD   DC.PRI          ;ARGUMENT FOR INTEN
                                                .ENDC
000004  004267  000714                          JSR     R2,CPTIS
                                        ;
                                        ;
                                        ;
000010                                  CPS2:
000010  0000006                                .WORD   CPRSR
000012     000                                  .BYTE   LNO
000013     100                                  .BYTE   100
000014  0000006                                .WORD   DE.PRI
000016  0000006                                .WORD   C.SPS
000020  0000006                                .WORD   DE.PRI
000022  0000006                                .WORD   C.SPS
000024  000076'                                .WORD   CRBL2+OFF
000026  000160'                                .WORD   CRBU2
000030  000046'                                .WORD   CRBIN2
000032  0000006                                .WORD   CFKLU
000034  000000                                 .WORD   0
000036  000000                                 .WORD   0
000040  000000                                 .WORD   0
000042  000000                                 .WORD   0
000044  000000                                 .WORD   0


000046                                  CPBS:           ;END OF ISR TABLE(S)
                                                .ENDC
                                                .IF DF RSTS
                                                .GLOBL  CPR2,CPT2,CPS2,CPBS,ICIN2,CSIN2,CFIN2,CRBLM2
                                                .ENDC
```

```
                                ;CIRCULAR RECEIVE BUFFER LOADER AND UNLOADER ROUTINES

                                ;THESE TWO ROUTINES PROVIDE A FACILITY FOR LOADING SINGLE
                                ;WORD ENTRIES INTO AND RETRIEVING SINGLE WORD ENTRIES FROM
                                ;AN INTERNALLY PROVIDED CIRCULAR RECEIVE BUFFER

                                ;BUFFER SIZE IS SPECIFIED AT ASSEMBLY TIME VIA THE PARAMETER CRBSZX

                                ;THE USER MUST DEFINE THE RETURN REGISTER RETNX, AND THE WORK
                                ;REGISTERS CBAX AND CBBX AT ASSEMBLY TIME (CBAX AND CBBX DEFINE WHICH
                                ;REGISTERS THE TAP MUST UTILIZE FOR CRB UNLOADER SERVICE.)

                                ;THE ROUTINES REQUIRE THAT BITS 15 AND 14 BE CLEARED BEFORE THE
                                ;LOADER IS CALLED, SINCE THESE ARE RESERVED FOR USE BY THE ROUTINES

                                ;TO CREATE A UNIQUE COPY OF THESE ROUTINES FOR A MULTI-CRB ENVIRONMENT
                                ;REPLACE X WHEREVER IT APPEARS AT THE END OF A SYMBOL
                                ;BY A UNIQUE CHARACTER
                                ;**********
                                ;CRB FOR DP11 LINE 2

                                ;CRB DEFINITIONS

                                        .IF NDF RSTS
000400                                  CRBSZ2=400              ;64 CHARACTER BUFFER
                                        .ENDC
000005                                  CBA2=%5
000004                                  CBB2=%4
000007                                  RETN2=%7

                                ;CRB INITIALIZATION ROUTINE
000046  016705  000102          CRBIN2: MOV     ICIN2,CBA2
000052  010567  000100                  MOV     CBA2,CSIN2
000056  010567  000166                  MOV     CBA2,CFIN2

000062  005065  000652'        CRBLP2: CLR     CRBLM2(CBA2)
000066  062705  000002                  ADD     #2,CBA2
000072  001373                          BNE     CRBLP2
000074  000207                          RTS     RETN2

                                ;CRB LOADER

000076  052704  100000          CPBL2:  BIS     #100000,CBB2
000102  016705  000050                  MOV     CSIN2,CBA2
                                        .IF DF RSX11D
                                        ADD     #OFF,CBA2
                                        .ENDC
000106  005765  000652'                 TST     CRBLM2(CBA2)
000112  100413                          BMI     CRBOV2
000114  010465  000652'                 MOV     CBB2,CRBLM2(CBA2)
000120  062767  000002  000030          ADD     #2,CSIN2
000126  001003                          BNE     CRVO2
000130  016767  000020  000020  CRBRS2. MOV     ICIN2,CSIN2
000136  000250                  CRVO2:  CLN
000140  000207                          RTS     RETN2
```

```
000142  052765  040000  000652' CRBOV2: BIS     #40000,CRBLM2(CBA2)
000150  000270                  SEN
000152  000207                  RTS     RETN2
                                .IF NDF RSTS
                        ;
                        ;
000154  177400          ICIN2:  .WORD   -CRBSZ2
000156  177400          CSIN2:  .WORD   -CRBSZ2
                                .ENDC
```

```
                                        ;CRB UNLOADER
                                        
000160  016705  000064          CRBU2:  MOV     CFIN2,CBA2
000164  016504  000652'                  MOV     CRBLM2(CBA2),CBB2
000170  100025                          BPL     CRBNO2
000172  005165  000652'                  COM     CRBLM2(CBA2)
000176  062767  000002  000044          ADD     #2,CFIN2
000204  001003                          BNE     CRYO2
000206  016767  177742  000034          MOV     ICIN2,CFIN2
000214  032704  040000          CRYO2:  BIT     #40000,CBB2
000220  001004                          BNE     CRBSV2
000222  042704  140000                  BIC     #140000,CBB2
000226  000252                          CLVN
000230  000207                          RTS     RETN2
                                        
000232  042704  140000          CRBSV2: BIC     #140000,CBB2
000236  000250                          CLN
000240  000262                          SEV
000242  000207                          RTS     RETN2
                                        
000244  000270                  CRBNO2: SEN
000246  000207                          RTS     RETN2
                                        
                                        .IF NDF RSTS
                                        ;
                                        ;
000250  177400                  CFIN2:  .WORD   -CRBSZ2
                                        
                                        ;BUFFER STORAGE FOR CRB#2.
                                        
        000652'                         .=.+CRBSZ2
        000652'                         CRBLM2=.                        ;BUFFER LIMIT
                                        .ENDC
                                        
                                        ;;;;;;;;;;END OF DP11.TBL;;;;;;;;;;
```

```
                                        ;DP11 ISR TAPE 2 OF 2
                                        ;DP11 ISR COMMON PROCESSING - RECEIVE
                                        ;ENTERED WITH R2 SAVED ON STACK VIA JSR R2, CPRIS
000652                                  CPRIS:
                                                .IF NDF RSTS    ;RSTS HAS ALREADY SAVED ALL REGS
000652  010346                                  MOV     R3,-(SP)        ;SAVE THREE MORE REGS
                                                .IF NDF RSX11M&RT11     ;$INTSV ALREADY HAS SAVED R4 & R5
                                                MOV     R4,-(SP)
                                                MOV     R5,-(SP)
                                                .ENDC
                                                .ENDC
000654  016204  000004                        MOV     SRA-R(R2),R4    ;GET RECVR STATUS ADDR
000660  005762  000036                        TST     SSC-R(R2)       ;TEST FOR CLR STP SYNC ON 1ST NON-SYNC
000664  100002                              BPL     CPRB1           ;NOT SET
000666  042714  000001                        BIC     #1,(R4)         ;CLEAR STRIP SYNC
000672  011403                        CPRB1:  MOV     (R4),R3         ;GET DP11 RECVR STATUS REG
000674  016404  000002                        MOV     2(R4),R4        ;GET DATA CHARACTER
000700  032703  010000                        BIT     #CPB12,R3       ;TEST FOR PARITY BIT SET
000704  001402                              BEQ     CPRB2           ;BR IF EVEN PARITY
000706  052704  010000                        BIS     #CPB12,R4       ;SET ODD PARITY CHECK BIT
000712  004772  000020                CPRB2:  JSR     PC,@CBL-R(R2)   ;GO LOAD CIRCULAR RECEIVE BUFFER
                                                .IF NDF RSTS
                                                .IF NDF RSX11M&RT11
                                                MOV     (SP)+,R5        ;RESTORE REGS
                                                MOV     (SP)+,R4
                                                .ENDC
000716  012603                                  MOV     (SP)+,R3
                                                .ENDC
000720  012602                                  MOV     (SP)+,R2
                                                .IF DF  RSX11D
                                                JMP     @#..INTX
                                                .ENDC
                                                .IF DF RSTS
                                                RETURN          ;
                                                .ENDC
                                                .IF DF RSX11M!RT11
000722  000207                                  RTS     PC              ;RETURN TO CODE IN THE EXEC BUFFER POOL
                                                .ENDC
```

```
                                        ;DP11 ISR COMMON PROCESSING - TRANSMIT
                                        ;ENTERED WITH R2 SAVED ON STACK VIA JSR R2,CPTIS
000724                                  CPTIS:
                                                .IF NDF RSTS                    ;RSTS HAS SAVED ALL THE REGS
000724  010346                                  MOV     R3,-(SP)                ;SAVE REGS
000726  010046                                  MOV     R0,-(SP)                ;
                                                .IF NDF RSX11M&RT11
                                                MOV     R4,-(SP)
                                                MOV     R5,-(SP)
                                                .ENDC
                                                .ENDC
000730  016204  000000                          MOV     SRA-X(R2),R4            ;GET RECVR STATUS ADDR
000734  016403  000004                          MOV     4(R4),R3                ;GET XMIT STATUS
000740  042703  013577                          BIC     #13577,R3               ;CLEAR ALL BUT INTERRUPT
                                                                                ;CAUSING FLAGS ...
000744  040364  000004                          BIC     R3,4(R4)                ;... IN TCSR
000750  032703  160000                          BIT     #160000,R3              ;TEST FOR ERROR
000754  001062                                  BNE     CPRER                   ;BR IF ERROR
000756  005362  000030                CPTIO:    DEC     XBC-X(R2)               ;DECR BYTE CNTR
000762  001423                                  BEQ     CPTB4                   ;CNTR NOW ZERO
000764  005262  000026                          INC     XBA-X(R2)               ;UPDATE BUFF ADDR POINTER
000770  032762  000200  000034                  BIT     #200,WBF-X(R2)          ;IS IT BYTE TYPE
000776  001004                                  BNE     CPTB2                   ;NO
001000  117264  000026  000006        CPTB1:    MOVB    @XBA-X(R2),6(R4)                ;OUTPUT THE CHAR
001006  000405                                  BR      CPTIX                   ;GO EXIT
001010  005262  000026                CPTB2:    INC     XBA-X(R2)               ;UPDATE BUFF ADDR POINTER
                                                                                ;TO NEXT WORD
001014  017264  000026  000006        CPTB3:    MOV     @XBA-X(R2),6(R4)                ;OUTPUT THE CHAR
001022                                  CPTIX:
                                                .IF NDF RSTS
                                                .IF NDF RSX11M&RT11
                                                MOV     (SP)+,R5                ;RESTORE REGS
                                                MOV     (SP)+,R4
                                                .ENDC
001022  012600                                  MOV     (SP)+,R0                ;
001024  012603                                  MOV     (SP)+,R3
                                                .ENDC
001026  012602                                  MOV     (SP)+,R2
                                                .IF DF  RSX11D
                                                JMP     @#..INTX
                                                .ENDC
                                                .IF DF  RSTS
                                                RETURN
                                                .ENDC
                                                .IF DF RSX11M!RT11
001030  000207                                  RTS     PC                      ;RETURN TO CODE IN THE EXEC BUFFER POOL
                                                .ENDC

001032  016203  000024          CPTB4:    MOV     XWP-X(R2),R3            ;WORD PAIR PTR
                                                .IF DF  RSX11D
                                                ADD     #0FF,R3
                                                .ENDC
001036  012362  000026                          MOV     (R3)+,XBA-X(R2) ;OUT IN CURR. XMIT ADDR PTR
001042  001411                                  BEQ     CPTB5                   ;IF ZERO, WE'RE DONE
                                                .IF DF  RSX11D
                                                ADD     #0FF,XBA-X(R2)
                                                .ENDC
```

```
 98 001044  012362  000030                    MOV     (R3)+,XBC-X(R2) ;PUT IN CURR. XMIT BYTE CNTR
 99                                           .IF DF  RSX11D
100                                           SUB     #0FF,R3
101                                           .ENDC
102
103
104 001050  010362  000024                    MOV     R3,XWP-X(R2)    ;PUT IN PTR TO NEXT WORD PAIR
105 001054  032762  000200  000034            BIT     #200,WBF-X(R2)  ;IS IT BYTE TYPE
106 001062  001354                            BNE     CPTB3           ;NO. GO OUTPUT CHAR
107 001064  000745                            BR      CPTB1           ;GO OUTPUT CHAR
108
109 001066                          CPTB5:
110                                           .IF DF RSX
111 001066  016700  000000G                   MOV     TASKAD,R0       ;
112                                           .IF DF RSX11D
113                                           BIS     #40,A.EF(R0)    ;SET 'SIB ENTRY EVENT FLAG'
114                                           BISB    #EV.SE,@#.SERFG         ;SET SIGNIFICANT EVENT
115                                           .IFF
116 001072  052760  000040  000000G           BIS     #40,T.EFLG(R0)  ;SET SIB ENTRY EVENT FLAG
117 001100  016767  000000G 000000G           MOV     $TSKHD,$RQSCH           ;DECLARE SIG EVENT
118 001106  005067  000000G                   CLR     $SIGFL          ;
119                                           .ENDC
120                                           .IFF
121                                           MOV     LNR-X(R2),R5    ;GET LINE NUMBER - CODE
122                                           ADD     #1000,R5        ;ADD 2 TO TYPE CODE
123                                           BIC     #40000,R5       ;1WORD WORK ENTRY
124                                           JSR     PC,@SBL-X(R2)   ;GO LOAD INTERFACE BUFFER
125                                           .ENDC
126 001112  042764  000100  000004            BIC     #100,4(R4)      ;CLEAR INTERRUPT ENABLE
127 001120  000740                            BR      CPTIX           ;NO EXIT
128                                 ;
129 001122  032703  040000          CPRER:    BIT     #CPB14,R3       ;TEST FOR DATA OVERRUN FLAG
130 001126  001410                            BEQ     CPER2           ;BR IF NOT SET
131 001130  016404  000002                    MOV     2(R4),R4        ;READ DATA REG FOR CHAR
132 001134  052704  020000                    BIS     #CPB13,R4       ;SET RECEIVER OVERRUN BIT
133 001140  004772  000014                    JSR     PC,@CBL-X(R2)   ;GO LOAD CRB
134 001144  016204  000000                    MOV     SRA-X(R2),R4    ;RESTORE STATUS POINTER
135
136
137 001150                          CPER2:
138 001150  105703                  CPER6:    TSTB    R3              ;TRANSMIT DONE ALSO?
139 001152  100701                            BMI     CPTIO           ;YES, GO PROCESS IT
140 001154  000722                            BR      CPTIX           ;NO. GO TO EXIT
141                                           .IF DF  RSX11D!RSX11M
142 000000                                     PSECT   XXX
143                                           .ENDC
```

```
                                        ;DP11 OUTPUT STARTUP ROUTINE
                                        ;ENTERED FROM MATRIX TYPE SWITCH WITH
                                        ;REGS SET AS FOLLOWS
                                        ;       R1 - POINTER TO INDICATOR WORD
                                        ;            FIRST X-MIT WORD PAIR FOLLOWS INDICATOR WORD
                                        ;       R2 - POINTER TO BASE OF LINE TABLE
                                        ;       R3 - LINE # AS A WORD INDEX
                                        ;IF ILLEGAL ADDRESS OR BYTE COUNT, RETURN
                                        ;WITH N CONDITION CODE SET

        000002                                  CPINX=2                 ;LINE TABLE INDEX TO ISR TABLE POINTER

000000  016203  000002          CP.OSU: MOV     CPINX(R2),R3    ;GET ISR TABLE PTR FOR LINE
000004  016305  000010                  MOV     SRA(R3),R5      ;SET POINTER TO RCSR
000010  012104                          MOV     (R1)+,R4        ;PUT INDICATOR WORD IN R4
000012  105704                          TSTB    R4              ;BYTE/WORD FLAG TO BE SET
000014  100012                          BPL     CPOSU2          ;NO
000016  032704  000100                  BIT     #100,R4         ;IS BIT 6 SET
000022  001404                          BEQ     CPOSU1          ;NO
000024  052763  000200  000044          BIS     #200,WBF(R3)    ;SET FOR WORD TYPE
000032  000403                          BR      CPOSU2
000034  042763  000200  000044  CPOSU1: BIC     #200,WBF(R3)    ;CLEAR FOR BYTE TYPE
000042  005704                  CPOSU2: TST     R4              ;IS IDLE SYNC TO BE CHANGED
000044  100012                          BPL     CPOSU4          ;NO
000046  032704  040000                  BIT     #40000,R4       ;IS BIT 14 SET
000052  001404                          BEQ     CPOSU3          ;NO
000054  052765  000002  000004          BIS     #2,4(R5)        ;SET IDLE SYNC
000062  000403                          BR      CPOSU4
000064  042765  000002  000004  CPOSU3: BIC     #2,4(R5)        ;CLEAR IDLE SYNC
000072  012104                  CPOSU4: MOV     (R1)+,R4        ;PUT BUFF ADDR PTR IN R4
000074  001426                          BEQ     CPOSUX          ;(IF ZERO, THIS WAS IDLE SYNC REQ. ONLY)
000076  010463  000036                  MOV     R4,XBA(R3)      ;AND IN CURR.XMIT ADDR PTR
000102  012102                          MOV     (R1)+,R2        ;PUT BUFF BYTE CNT PTR IN R2
000104  003424                          BLE     CPOSU7          ;ILLEGAL BYTE COUNT
000106  010263  000040                  MOV     R2,XBC(R3)      ;PUT IN CURR.XMIT CNTR PTR
000112  010163  000034                  MOV     R1,XWP(R3)      ;PUT IN PTR TO NEXT XMIT WORD PAIR
000116  032763  000200  000044  CPOSU9: BIT     #200,WBF(R3)    ;IS IT BYTE TYPE
000124  001004                          BNE     CPOSU5          ;NO
000126  117365  000036  000006          MOVB    @XBA(R3),6(R5)  ;OUTPUT 1ST CHAR
000134  000403                          BR      CPOSU6
000136  017365  000036  000006  CPOSU5: MOV     @XBA(R3),6(R5)  ;OUTPUT 1ST CHAR
000144                          CPOSU6:
                                        .IF DF  RSX11D
                                        ADD     #0FF,XBA(R3)
                                        .ENDC
000144  052765  000100  000004          BIS     #100,4(R5)              ;SET INTERRUPT ENABLE
000152  000250                  CPOSUX: CLN
000154  000207                          RTS     PC              ;MATRIX SWITCH EXIT

000156  000270                  CPOSU7: SEN
000160  000207                          RTS     PC              ;MATRIX SWITCH EXIT
```

```
                                        ;DP11 MODEM CONTROL ROUTINE
                                        ;ENTERED FROM DEVICE TYPE MATRIX SWITCH WITH
                                        ;REGS SET AS FOLLOWS
                                        ;       R1 - POINTER TO THREE WORD PARAMETER LIST
                                        ;       R2 - POINTER TO BASE OF LINE TABLE
                                        ;       R3 - LINE # AS A WORD INDEX
                                        ;IF MODEM STATUS REQUESTED, STATUS WORD IS PLACED IN
                                        ;THIRD WORD OF USER PARAMETERS
                                        ;IF ILLEGAL OPERATION, RETURN WITH N CONDITION CODE SET

                                        ;FORMAT OF PARAMETER LIST
                                        ;LOCP:     UNSPECIFIED
                                        ;LOCP+2:   MODEM CONTROL BIT SETTINGS
                                        ;LOCP+4:   FOR MODEM STATUS RETURN

000162  016104  000002         CP.MDM: MOV     2(R1),R4        ;R4 HAS BIT SETTINGS
000166  016205  000002                 MOV     2(R2),R5        ;R5 HAS ISR TABLE ADDR FOR LINE
000172  016503  000010                 MOV     SRA(R5),R3      ;R3 HAS RECVR STATUS REG ADDR
000176  010305                         MOV     R3,R5           ;COPY R3 INTO R5
000200  062705  000004                 ADD     #4,R5           ;R5 HAS XMIT STATUS REG ADDR
000204  032704  100125                 BIT     #100125,R4      ;IS IT PROPER COMMAND
000210  001415                         BEQ     CPMER           ;NO-ITS ERROR
000212  032704  000001                 BIT     #1,R4           ;IS BIT 0 TERM RDY IND SET
000216  001405                         BEQ     CPMCA           ;NO
000220  032704  000002                 BIT     #2,P4           ;IS BIT 1 EQUAL 1
000224  001411                         BEQ     CPMRA           ;NO-GO CLEAR TERM READY
000226  052715  000001                 BIS     #1,(R5)         ;SET TERMINAL READY
000232                          CPMCA:
000232  032704  100000          CPMCD: BIT     #100000,R4      ;BIT 15 SET TO READ STATUS
000236  001012                         BNE     CPMCG           ;YES-GO READ STATUS
000240  000254                 CPMGE:  CLZN                    ;SET N AND Z TO 0
000242  000207                         RTS     PC              ;GO EXIT

000244  000270                 CPMER:  SEN                     ;SET N TO 1
000246  000207                         RTS     PC              ;GO EXIT

000250  042715  000001         CPMRA: BIC     #001,(R5)               ;CLEAR TERM READY
000254  042765  004100  177774         BIC     #4100,-4(R5)    ;CLEAR RECEIVE ENABLE AND ACTIVE
000262  000763                         BR      CPMCA
                                ;SENSE MODEM CONTROL

000264  005004                 CPMCG:  CLR     R4              ;CLEAR R4 TO BUILD SENSE WORD
000266  011303                         MOV     (R3),R3         ;READ RCSR
000270  011505                         MOV     (R5),R5         ;READ TCSR
000272  032705  001000                 BIT     #1000,R5        ;IS REQ TO SEND SET
000276  001402                         BEQ     CPMCH           ;NO-NOT SET
000300  052704  000001                 BIS     #1,R4           ;SET REQ TO SEND ON
000304  032705  002000         CPMCH:  BIT     #2000,R5        ;IS CLEAR TO SEND SET
000310  001402                         BEQ     CPMCJ           ;NO
000312  052704  000002                 BIS     #2,R4           ;SET CLR TO SEND ON
000316  032705  010000         CPMCJ:  BIT     #10000,R5       ;IS DATA SET READY SET
000322  001402                         BEQ     CPMCK           ;NO
000324  052704  000004                 BIS     #4,R4           ;SET DATA SET READY
000330  032705  000001         CPMCK:  BIT     #1,R5           ;IS DATA TERM READY SET
000334  001402                         BEQ     CPMCL           ;NO-NOT SET
000336  052704  000010                 BIS     #10,R4          ;SET DTR IND
000342                          CPMCL:
```

```
254
255 000342  032705  004000      CPMCM:  BIT     #4000,R5        ;IS CARRIER ON
256 000346  001402                      BEQ     CPMCN           ;NO
257 000350  052704  000040              BIS     #40,R4          ;SET CARRIER ON
258 000354                      CPMCN:
259 000354  010461  000004      CPMCP:  MOV     R4,4(R1)        ;STORE SENSE WORD
260 000360  000727                      BR      CPMGE           ;NOW GOOD EXIT
```

```
                                .IF DF RSX
001156                          .PSECT  ISR
                                .ENDC
                        ;DP11 MODE CONTROL ROUTINE
                        ;ENTERED FROM DEVICE TYPE MATRIX SWITCH WITH
                        ;REGS SET AS FOLLOWS
                        ;       R1 - POINTER TO FOUR WORD PARAMETER LIST
                        ;       R2 - POINTER TO BASE OF LINE TABLE
                        ;       R3 - LINE # AS A WORD INDEX

                        ;FORMAT OF PARAMETER LIST
                        ;LOCP:      UNSPECIFIED (LOW BYTE)
                        ;LOCP+1:    40=SENSE REQUESTED
                        ;LOCP+2:    MODE CONTROL BIT SETTINGS
                        ;LOCP+4:    FOR MODE STATUS RETURN
                        ;LOCP+6:    SYNC CHARACTER WORD

001156  016104  000002  CP.MOD:  MOV     2(R1),R4        ;R4 HAS BIT SETTINGS
001162  016202  000002          MOV     2(R2),R2        ;R2 HAS CPR1 ADDR FOR DP11 LINE
001166  016205  000010          MOV     SRA(R2),R5      ;R5 HAS RECVR STATUS REG ADDR
001172  032704  115041          BIT     #115041,R4      ;IS IT SET MODE COMMAND
001176  001471                  BEQ     CPCEX           ;NO-GO SEE IF SENSE
                                                        ;GO EXIT - SINCE DON'T DO SENSE
001200  032704  000001          BIT     #1,P4           ;IS INT AND M IND SET
001204  001405                  BEQ     CPCA            ;NO-GO TEST NEXT
001206  032704  000010  CPCGA:  BIT     #10,R4          ;IS RCVE INT ENAB IND SET
001212  001447                  BEQ     CPCH            ;NO
001214  052715  000100          BIS     #100,(R5)       ;SET RCVE INT ENAB IN RCSR
001220                  CPCHA:
001220                  CPCA:
001220  032704  001000  CPCB:   BIT     #1000,R4        ;IS HALF DUPLEX IND SET
001224  001405                  BEQ     CPCC            ;NO-GO TEST FOR NEXT
001226  032704  002000          BIT     #2000,R4        ;BIT 10 SET
001232  001442                  BEQ     CPCM            ;NO
001234  052715  000002          BIS     #2,(R5)         ;SET BIT 1 IN RCSR FOR HALF DUPLEX
001240  032704  004000  CPCC:   BIT     #4000,R4        ;IS SYNC CONTROL IND SET
001244  001406                  BEQ     CPCD            ;NO-GO TEST FOR NEXT
001246  116165  000006  000003  MOVB    6(R1),3(R5)     ;MOVE LOW BYTE OF SYNC CHAR
001254  116165  000007  000007  MOVB    7(R1),7(R5)     ;MOVE HIGH BYTE OF SYNC CHAR
001262  032704  010000  CPCD:   BIT     #10000,R4       ;BIT 12 SET(STRIP SYNC IND)
001266  001435                  BEQ     CPCS            ;NO-GO TEST FOR SENSE
001270  032704  020000          BIT     #20000,R4       ;BIT 13 SET
001274  001424                  BEQ     CPCP            ;NO
001276  052715  000001          BIS     #1,(R5)         ;SET BIT 0 IN RCSR
001302  032704  040000  CPCPA:  BIT     #40000,R4       ;BIT 14 SET
001306  001422                  BEQ     CPCR            ;NO
001310  052762  177777  000042  BIS     #-1,SSC(R2)     ;SAVE CLR STRIP SYNC IND
001316  032704  100000  CPCPB:  BIT     #100000,R4      ;RECEIVE ACTIVE TO BE CLEARED
001322  001417                  BEQ     CPCS            ;NO
001324  042715  004000          BIC     #4000,(R5)              ;YES. CLEAR IT
001330  000414                  BR      CPCS            ;GO TEST FOR SENSE
001332  042715  000100  CPCH:   BIC     #100,(R5)       ;CLEAR RCVE INT ENAB IN RCSR
001336  000730                  BR      CPCHA
001340  042715  000002  CPCM:   BIC     #2,(R5)         ;CLEAR BIT 1 IN RCSR
001344  000735                  BR      CPCC
001346  042715  000001  CPCP:   BIC     #1,(R5)         ;CLEAR BIT 0 IN RCSR
```

```
001352  000753                          BR      CPCPA
001354  005062  000042         CPCR:    CLR     SSC(R2)         ;SAVE DON'T CLR STRIP SYNC IND
001360  000756                          BR      CPCPB

                                ;SENSE MODE CONTROL

001362                          CPCS:
001362  000254                  CPCEX:  CLZN                    ;SET GOOD EXIT CODE
001364  000207                          RTS     PC              ;RETURN TO USER
                                ;
                                ;
                                        .IF DF RSX
000362                                  .PSECT  XXX
                                        .ENDC
                                ;
                                ;
                                ;DP11 KILL I/O ROUTINE
                                ;ENTERED FROM DEVICE TYPE MATRIX SWITCH
                                ;WITH REGISTERS PRESET AS FOLLOWS:

                                ;       R1-POINTER TO KILL I/O PARAMETER LIST
                                ;       R2-POINTER TO BASE OF TAP LINE TABLE
                                ;       R3-LINE # AS A WORD INDEX

000362                          CP.KLL:
000362  000270                  CP.KER: SEN                             ;INDICATE ERROR
000364  000207                          RTS     PC
```

```
                                ;DP11 LINE TABLE INITIALIZATION ROUTINE
                                ;ENTERED FROM DEVICE TYPE MATRIX SWITCH WITH
                                ;REGS SET AS FOLLOWS
                                ;       R1 - POINTER TO USER PARAMETER LIST
                                ;       R2 - POINTER TO BASE OF LINE TABLE
                                ;       R3 - LINE # AS A WORD INDEX
                                ;REPORTS BACK ERROR CODE 4 IF LINE NUMBER NOT FOUND
                                        .GLOBL  CPAS


000366                          CP.INI:
000366  012700  000000'                MOV     #CPAS,R0        ;START OF DP11 LINES
000372  010062  000002         CPTINB: MOV     R0,2(R2)        ;SET UP DP11 ADDR
000376  016062  000026  000004         MOV     CBU(R0),4(R2)   ;SET POINTER TO CRB UNLOADER
000404  004770  000030                 JSR     PC,@CBI(R0)     ;CALL CRB INIT ROUTINE
                                        .IF DF RSTS
                                        CMP     SRA(R0),#170000 ;HAS THE POINTER THAT THE MONITOR GIVES US
                                                                ;BEEN CHANGED TO THE ADDRESS?
                                        BHI     CPTIN1          ;YES
                                        MOV     @SRA(R0),R3     ;NO - SET IT UP RIGHT FOR THIS CODE
                                        MOV     R3,SRA(R0)      ;
                                        CMP     SRA(R0),#170000 ;IS CSR.RJ SET RIGHT?
                                        BLO     ERROR           ;NO - IS ERROR
                                CPTIN1:
                                        .ENDC
                                        .IF DF  RSX11D!RSX11M!RT11
000410  010005                        MOV     R0,R5           ;COPY TABLE ADDRESS
000412  012700  000000G               MOV     #CPVP,R0
000416  012701  000000'               MOV     #CPR2,R1
000422  005002                        CLR     R2
000424  016503  000020                MOV     RPR(R5),R3
000430  004737  000000G               JSR     PC,@#..CINT     ;CONNECT TO INTERRUPT VECTOR
000434  103424                        BCS     ERROR
000436  022020                        CMP     (R0)+,(R0)+     ;UP TO TRANSMIT VECTOR
000440  012701  000004'               MOV     #CPT2,R1
000444  016503  000014                MOV     XPR(R5),R3
000450  004737  000000G               JSR     PC,@#..CINT     ;CONNECT TO TRANSMIT VECTOR
000454  103414                        BCS     ERROR
000456  012700  000000'               MOV     #CPAS,R0        ;RESET START ADDR CAUSE CLOBBERED IT
                                        .ENDC
000462  016003  000010                MOV     SRA(R0),R3      ;NOW GET STATUS REGISTER
000466  005023                        CLR     (R3)+           ;AND CLEAR IT
000470  005723                        TST     (R3)+
000472  005023                        CLR     (R3)+           ;
                                        .IF DF  RT11
                                        MOV     #AT.END,R0      ;SET UP FOR EXIT TIME
                                        .DEVICE                 ;DISABLE INTERRUPT WHEN THE
                                                                ;PROGRAM ABORTS FOR ANY REASON
                                        .ENDC
000474  000250                        CLN
000476  000207                CPTIND: RTS     PC              ;MATRIX SWITCH EXIT
000500  052761  000010  000004  CPTINC: BIS     #10,4(R1)       ;SET USER ERROR CODE
000506  000270                ERROR:  SEN
000510  000207                        RTS PC                  ;NOW EXIT

                                         IF DF RT11
                                ;;;;;;;;;;;;;;;;;;;;
```

```
                        ; PARAMETER BLOCK FOR .DEVICE CALL
AT.END: .BYTE   0               ;NO CHANNEL
        .BYTE   14              ;FUNCTION CODE
        .WORD   ATTEND          ;LIST OF WORDS TO BE LOADED
                                ;AT PROGRAM TERMINATION
;
;
ATTEND: .WORD   CPRSR           ;ADDRESS OF COMM CSW
        .WORD   0               ;FORCE IT TO 0
        .WORD   CPRSR+4         ;TRANSMIT CSW
        .WORD   0               ;ALSO FORCE TO 0
        .WORD   0               ;END OF LIST
;
;;;;;. . . . . ;
        .END
```

```
                                        ;DP-11 RESTART  ROUTINE
                                        ;ENTERED FROM DEVICE TYPE MATRIX SWITCH WITH
                                        ;REGS SET AS FOLLOWS:
                                        ;       R1 - POINTER TO USER PARAMETER LIST
                                        ;       R2 - POINTER TO BASE OF LINE TABLE
                                        ;       R3 - LINE # AS A WORD INDEX


                                                .GLOBL  CP.RST


000512                                  CP.RST:
                                                .IF NDF RSTS
000512  012705  000000'                         MOV     #CPAS,R5        ;GET ADDR OF START OF TABLE
                                                .IF DF RSX11D!RSX11M!RT11
000516  012700  000000G                         MOV     #CPVP,R0        ;SET UP TO DISCONNECT FROM VECTOR
000522  004737  000000G                         JSR     PC,@#..DINT     ;AND DO IT
000526  022020                                  CMP     (R0)+,(R0)+     ;THEN BUMP TO NEXT VECTOR ADDRESS
000530  004737  000000G                         JSR     PC,@#..DINT     ;AND DISCONNECT FROM IT
                                                .ENDC
000534  016503  000010                         MOV     SRA(R5),R3      ;AND THEN ADD OF STATUS REG
000540  005023                                  CLR     (R3)+           ;CLEAR THE REGISTER
000542  005723                                  TST     (R3)+           ;
000544  005023                                  CLR     (R3)+           ;
                                                .ENDC
000546  000250                                  CLN
000550  000207                                  RTS     PC
                                                .IF DF RSTS

                                                TMPORG  RJMINT,34
                                                .WORD   CTPLR
                                                UNORG
                                        ;
                                        GLOBAL  <CTPLR>
                                        ;
                                                TMPORG  RJMCTL,12
                                                .WORD   CPRIS
                                                UNORG
                                        ;
                                                TMPORG  RJMCTL,16
                                                .WORD   CPTIS
                                                UNORG
                                        ;
                                        GLOBAL  <CPRIS,CPTIS>
                                        ;
                                                TMPORG  RJMCTL,34
                                                .WORD   CRBL2
                                                .WORD   CRBU2
                                                .WORD   CRBIN2
                                                .WORD   CFKLU
                                        ;
                                        GLOBAL  <CRBL2,CRBU2,CRBIN2,CFKLU>
                                        ;

                                                DEFORG  RJEPAT
                                        RJEPAT: .BLKW   2
                                                UNORG
```

```
                                ;
                                        .ENDC
                                ;
                                ;*****************************

000001                                  .END
```

```
DPV112  MACRO M1104  06-DEC-77 08:20  PAGE 16-2
SYMBOL TABLE

CBA2  =%000005          CPCPA   001302R      002 CPRB1   000672R      002 CRBIN2  000046R      002 RPR    = 000020
CBB2  =%000004          CPCPB   001316R      002 CPRB2   000712R      002 CRBLM2= 000652R      002 RSX    = 000001
CBI   = 000030          CPCR    001354R      002 CPRER   001122R      002 CRBLP2  000062R      002 RSX11M= 000001
CBL   = 000024          CPCS    001362R      002 CPRIS   000652R      002 CRBL2   000076R      002 R6     =%000006
CBU   = 000026          CPD26 = 000001           CPRSR = ****** G        CRBNO2  000244R      002 R7     =%000007
CCC   = 000257          CPER2   001150R      002 CPRTC = 000000          CRBOV2  000142R      002 SBL    = 000032
CFIN2   000250R     002 CPER6   001150R      002 CPR2    000000RG     002 CRBRS2  000130R      002 SCC    = 000277
CFKLU = ****** GX       CPINX = 000002           CPS2    000010R      002 CRBSV2  000232R      002 SEC    = 000261
CLC   = 000241          CPMCA   000232R      003 CPTB1   001000R      002 CRBSZ2= 000400           SEN    = 000270
CLN   = 000250          CPMCD   000232R      003 CPTB2   001010R      002 CRBU2   000160R      002 SEV    = 000262
CLV   = 000242          CPMCG   000264R      003 CPTB3   001014R      002 CRVO2   000136R      002 SEZ    = 000264
CLVN  = 000252          CPMCH   000304R      003 CPTB4   001032R      002 CRYO2   000214R      002 SEZN   = 000274
CLZ   = 000244          CPMCJ   000316R      003 CPTB5   001066R      002 CSIN2   000156R      002 SRA    = 000010
CLZN  = 000254          CPMCK   000330R      003 CPTINB  000372R      003 C.SARR= ****** G         SSC    = 000042
CPAS  = 000000RG    002 CPMCL   000342R      003 CPTINC  000500R      003 C.SPS = ****** G         STC    = 000013
CPBS    000046R     002 CPMCM   000342R      003 CPTIND  000476R      003 DE.PRI= ****** G         TASKAD= ****** GX
CPB12 = 010000          CPMCN   000354R      003 CPTIS   000724R      002 ERROR   000506R      003 T.EFLG= ****** GX
CPB13 = 020000          CPMCP   000354R      003 CPTIX   001022R      002 HALT  = 000000           WBF    = 000044
CPB14 = 040000          CPMER   000244R      003 CPTIO   000756R      002 ICIN2   000154R      002 X      = 000010
CPB15 = 100000          CPMGE   000240R      003 CPTL  = 000046           LNO   = 000000           XBA    = 000036
CPCA    001220R     002 CPMRA   000250R      003 CPT2    000004RG     002 LNR   = 000012           XBC    = 000040
CPCB    001220R     002 CPOSUX  000152R      003 CPVP  = ****** G         NOP   = 000240           XBR    = 000016
CPCC    001240R     002 CPOSU1  000034R      003 CP.INI  000366RG     003 OFF   = 000000           XPR    = 000014
CPCD    001262R     002 CPOSU2  000042R      003 CP.KER  000362R      003 OS    = 000000           XWP    = 000034
CPCEX   001362R     002 CPOSU3  000064R      003 CP.KLL  000362RG     003 PS    = 177776           $RQSCH= ****** GX
CPCGA   001206R     002 CPOSU4  000072R      003 CP.MDM  000162RG     003 R     = 000004           $SIGFL= ****** GX
CPCH    001332R     002 CPOSU5  000136R      003 CP.MOD  001156RG     002 RBR   = 000022           $TSKHD= ****** GX
CPCHA   001220R     002 CPOSU6  000144R      003 CP.OSU  000000RG     003 RCVSIB= 000000           ..CINT= ****** GX
CPCM    001340R     002 CPOSU7  000156R      003 CP.RST  000512RG     003 RETN2 =%000007           ..DINT= ****** GX
CPCP    001346R     002 CPOSU9  000116R      003 CRBINT= 000000

. ABS.    000000    000
          000000    001
ISR       001366    002
XXX       000552    003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  803 WORDS  ( 4 PAGES)
DYNAMIC MEMORY:  2810 WORDS  ( 10 PAGES)
ELAPSED TIME:  00:00:14
 256,020 DP11;27, 256,030 DP11/CRF/-SP=ISRRXM,OPDEF,DP11.TBL,DP11.ISR
```

DP11 CREATED BY MACRO ON 6-DEC-77 AT 08:20 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES								
CBA2	=%000005		#7-261	*7-266	7-267	7-268	7-270	*7-271	*7-278	7-282	7-284
			7-291	*8-304	8-305	8-307					
CBB2	=%000004		#7-262	*7-277	7-284	*8-305	8-311	*8-313	*8-317		
CBI	= 000030		#5-170	5-171	15-361						
CBL	= 000024		#5-168	5-169	10-22	11-133					
CBU	= 000026		#5-169	5-170	15-360						
CCC	= 000257		#2-10								
CFIN2	000250	R	*7-268	8-304	*8-308	*8-310	#8-328				
CFKLU	= ******	GX	6-220								
CLC	= 000241		#2-6								
CLN	= 000250		#2-9								
CLV	= 000242		#2-7								
CLVN	= 000252		#3-118	8-314							
CLZ	= 000244		#2-8								
CLZN	= 000254		#3-116	13-227	14-326						
CPAS	= 000000	RG	#6-182	15-354	15-358	15-385	16-433				
CPBS	000046	R	#6-228								
CPB12	= 010000		#4-135	10-19	10-21						
CPB13	= 020000		#4-136	11-132							
CPB14	= 040000		#4-137	11-129							
CPB15	= 100000		#4-138								
CPCA	001220	R	14-287	#14-292							
CPCB	001220	R	#14-293								
CPCC	001240	R	14-294	#14-298	14-317						
CPCD	001262	R	14-299	#14-302							
CPCEX	001362	R	14-284	#14-326							
CPCGA	001206	R	#14-288								
CPCH	001332	R	14-289	#14-314							
CPCHA	001220	R	#14-291	14-315							
CPCM	001340	R	14-296	#14-316							
CPCP	001346	R	14-305	#14-318							
CPCPA	001302	R	#14-307	14-319							
CPCPB	001316	R	#14-310	14-321							
CPCR	001354	R	14-308	#14-320							
CPCS	001362	R	14-303	14-311	14-313	#14-325					
CPD26	= 000001		#3-83								
CPER2	001150	R	11-130	#11-137							
CPER6	001150	R	#11-138								
CPINX	= 000002		#12-155	12-157							
CPMCA	000232	R	13-220	#13-224	13-235						
CPMCD	000232	R	#13-225								
CPMCG	000264	R	13-226	#13-238							
CPMCH	000304	R	13-242	#13-244							
CPMCJ	000316	R	13-245	#13-247							
CPMCK	000330	R	13-248	#13-250							
CPMCL	000342	R	13-251	#13-253							
CPMCM	000342	R	#13-255								
CPMCN	000354	R	13-256	#13-258							
CPMCP	000354	R	#13-259								
CPMER	000244	R	13-218	#13-230							
CPMGE	000240	R	#13-227	13-260							
CPMRA	000250	R	13-222	#13-233							

SYMBOL		VALUE		REFERENCES						
CPOSUX		000152	R	12-175	#12-191					
CPOSU1		000034	R	12-163	#12-166					
CPOSU2		000042	R	12-161	12-165	#12-167				
CPOSU3		000064	R	12-170	#12-173					
CPOSU4		000072	R	12-168	12-172	#12-174				
CPOSU5		000136	R	12-182	#12-185					
CPOSU6		000144	R	12-184	#12-186					
CPOSU7		000156	R	12-178	#12-194					
CPOSU9		000116	R	#12-181						
CPRB1		000672	R	10-15	#10-17					
CPRB2		000712	R	10-20	#10-22					
CPRER		001122	R	11-58	#11-129					
CPRIS		000652	R	6-198	#10-5					
CPRSR	=	******	G	4-126	6-210					
CPRTC	=	000000		#4-133						
CPR2		000000	RG	3-69	#6-191	15-375				
CPS2		000010	R	#6-209						
CPTB1		001000	R	#11-64	11-107					
CPTB2		001010	R	11-63	#11-66					
CPTB3		001014	R	#11-68	11-106					
CPTB4		001032	R	11-60	#11-89					
CPTB5		001066	R	11-94	#11-109					
CPTINB		000372	R	#15-359						
CPTINC		000500	R	#15-398						
CPTIND		000476	R	#15-397						
CPTIS		000724	R	6-205	#11-43					
CPTIX		001022	R	11-65	#11-69	11-127	11-140			
CPTIO		000756	R	#11-59	11-139					
CPTL	=	000046		#5-177						
CPT2		000004	RG	3-69	#6-199	15-381				
CPVP	=	******	G	4-127	15-374	16-435				
CP.INI		000366	RG	3-70	#15-357					
CP.KER		000362	R	#14-344						
CP.KLL		000362	RG	3-70	#14-343					
CP.MDM		000162	RG	3-70	#13-212					
CP.MOD		001156	RG	3-70	#14-280					
CP.OSU		000000	RG	3-70	#12-157					
CP.RST		000512	RG	16-428	#16-431					
CRBINT	=	000000		#1-5						
CRBIN2		000046	R	6-219	#7-266					
CRBLM2	=	000652	R	*7-270	7-282	*7-284	*7-291	8-305	*8-307	#8-333
CRBLP2		000062	R	#7-270	7-272					
CRBL2		000076	R	6-217	#7-277					
CRBNO2		000244	R	8-306	#8-322					
CRBOV2		000142	R	7-283	#7-291					
CRBRS2		000130	R	#7-287						
CRBSV2		000232	R	8-312	#8-317					
CRBSZ2	=	000400		#7-259	7-297	7-298	8-328	8-332		
CRBU2		000160	R	6-218	#8-304					
CRVO2		000136	R	7-286	#7-288					
CRYO2		000214	R	8-309	#8-311					
CSIN2		000156	R	*7-267	7-278	*7-285	*7-287	#7-298		

SYMBOL	VALUE		REFERENCES								
C.SARR	= ******	G	3-68								
C.SPS	= ******	G	3-68	6-214	6-216						
DE.PRI	= ******	G	4-128	6-213	6-215						
ERROR	000506	R	15-379	15-384	#15-399						
HALT	= 000000		#2-4								
ICIN2	000154	R	7-266	7-287	#7-297	8-310					
LNO	= 000000		#3-81	6-211							
LNR	= 000012		#5-162	5-163							
NOP	= 000240		#2-5	#3-115							
OFF	= 000000		#4-143	6-217							
OS	= 000000		#3-58	3-80							
PS	= 177776		#3-103								
R	= 000004		#5-158	10-13	10-14	10-22					
RBR	= 000022		#5-167	5-168							
RCVSIB	= 000000		#1-4								
RETN2	=%000007		#7-263	*7-273	*7-289	*7-293	*8-315	*8-320	*8-323		
RPR	= 000020		#5-166	5-167	15-377						
RSTS	= ******		3-60	3-76	4-130	6-181	6-230	7-258	7-294	8-325	10-6
			10-23	10-34	11-44	11-70	11-82	15-362	16-432	16-447	
RSX	= 000001		#1-2	11-110	14-263	14-330					
RSX11D	= ******		3-57	4-125	4-139	7-279	10-31	11-79	11-90	11-95	11-99
			11-112	11-141	12-187	15-372	16-434				
RSX11M	= 000001		#1-3	3-57	4-125	4-145	10-8	10-24	10-37	11-47	11-71
			11-85	11-141	15-372	16-434					
RT11	= ******		3-57	3-71	4-125	6-192	6-200	10-8	10-24	10-37	11-47
			11-71	11-85	15-372	15-391	15-402	16-434			
R6	=%000006		#3-110								
R7	=%000007		#3-111								
SBL	= 000032		#5-171	5-172							
SCC	= 000277		#2-15								
SEC	= 000261		#2-11								
SEN	= 000270		#2-14								
SEV	= 000262		#2-12								
SEZ	= 000264		#2-13								
SEZN	= 000274		#3-117								
SRA	= 000010		#5-161	5-162	10-13	11-52	11-134	12-158	13-214	14-282	15-387
			16-440								
SSC	= 000042		#5-175	5-176	10-14	*14-309	*14-320				
STC	= 000013		#5-163	5-164							
TASKAD	= ******	GX	11-111								
T.EFLG	= ******	GX	*11-116								
WBF	= 000044		#5-176	5-177	11-62	11-105	*12-164	*12-166	12-181		
X	= 000010		#5-159	5-161	11-52	*11-59	*11-61	11-62	11-64	*11-66	11-68
			11-89	*11-93	*11-98	*11-104	11-105	11-133	11-134		
XBA	= 000036		#5-173	5-174	*11-61	11-64	*11-66	11-68	*11-93	*12-176	12-183
			12-185								
XBC	= 000040		#5-174	5-175	*11-59	*11-98	*12-179				
XBR	= 000016		#5-165	5-166							
XPR	= 000014		#5-164	5-165	15-382						
XWP	= 000034		#5-172	5-173	11-89	*11-104	*12-180				
$RQSCH	= ******	GX	*11-117								
$SIGFL	= ******	GX	*11-118								

DP11 CREATED BY MACRO ON 6-DEC-77 AT 08:20 PAGE 4
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES	
$TSKHD	= ******	GX	11-117	
..CINT	= ******	GX	15-378	15-383
..DINT	= ******	GX	16-436	16-438

```
DUV209  MACRO M1104  06-DEC-77 08:20  PAGE 1


                                        ;PREFIX FILE FOR RSX-11M DP11 OR DU11 ISR
000001                                  RSX=    1       ;SET FOR CUTBACK CODE JAN '77
000001                                  RSX11M=1
000000                                  RCVSIB=0
000000                                  CRBINT=0
```

```
                                ;PREFIX FILE FOR RSX11M ASSEMBLER AND ANY OTHER ASSEMBLER
                                ;WHICH DOES NOT RECOGNIZE OPCODE MNEMONICS EXCEPT WHEN THEY
                                ;APPEAR  IN THE OPCODE FIELD
000000                          HALT=0
000240                          NOP=240
000241                          CLC=241
000242                          CLV=242
000244                          CLZ=244
000250                          CLN=250
000257                          CCC=257
000261                          SEC=261
000262                          SEV=262
000264                          SEZ=264
000270                          SEN=270
000277                          SCC=277
```

```
                        ;
                        ;
                        ;
                        ; COPYRIGHT (C)  1975, 1976, 1977
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                        ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                        ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                        ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                        ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                        ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                        ;
                        ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ; DEC.
                        ;
                        ;COMTEX-11 DU11 ISR TABLES
                        ;VERSION V01-01 JLH 10/31/73
                        ;VERSION V01-02 JLH 11/15/73
                        ;VERSION V01-03 JLH 1/28/74
                        ;VERSION V02-01 HMS 4/1/74
                        ;VERSION V02-02 JLH 7/23/74
                        ;CORRECTS UNDEFINED IN RSTS ASSEMBLY
                        ;VERSION V02-03 RHR 9/4/74
                        ;CHANGES TO ASSEMBLE FOR RSX-11M
                        ;VERSION V02-04 RHR 7/2/75
                        ;MAKE DEVICE PRIORITY A GLOBAL DEFINITION FOR RSX
                        ;VERSION V02-05 RHR 9/5/75
                        ;MODIFICATIONS TO RUN WITH RT11 F/B
                        ;VERSION V02-06 RHR 12/11/75
                        ;MODIFICATIONS TO RUN THE DUP-11 IN "DU MODE"
                        ;VERSION V02-07 CEP 11/76
                        ;MODIFICATIONS TO RUN WITH RSTS V6B
                        ;VERSION V02-08 CEP 6/77
                        ;MODIFICATIONS TO RUN WITH LSI11
                        ;VERSION V02-09 CEP 9/77
                        ;MODIFICATIONS TO RUN CUT-CODE VERSION
                        ;
                        ;INTERNALLY DEFINED GLOBALS
                        .GLOBL CU.FST,CU.LST
                        ;
                        ;EXTERNALLY REFERENCED GLOBALS
                        .IF DF  RT11
                        
                        .GLOBL  CFKLU
                        .GLOBL  ..CINT,..DINT,RMON
                        .MCALL  .DEVICE,..V2..
                                ..V2..
                        .ENDC
                         IF DF  RSTS
```

```
                                .GLOBL  CFKLU,CSR.RJ
                                .ENDC
                                ;
                                ;2780 RCS DEFINITION.  DEFAULTS TO USE IN 2780 EMULATION
                                .IF NDF RCS
                                RCS=    1                       ;RCS=1 IMPLIES USE IN 2780 EMULATOR
                                .ENDC

                                        .IF DF  RSTS
                                        OS=0
                                        .ENDC
                                        .IF DF  RSX11D!RSX11M!RT11
        000000                          OS=0
                                        .ENDC
                                        .IF NDF OS
                                        LNO=2
                                        .ENDC
                                        .IF DF  OS
                                        .IF NDF RSTS
                                        .GLOBL  DUCSR           ;DEVICE ADDRESS EXTERNALLY DEFINED
                                        .ENDC
        000001                          VECTOR=1
        000000                          LNO=0
                                        .ENDC
                                ;VECTOR DEFINITION PROCESS.  DEFAULTS TO USE TABLES
                                .IF NDF VECTOR
                                VECTOR= 0                       ;USE VECTOR DEFINITION VIA .ASECT
                                .ENDC
```

```
                                .IF Z VECTOR
                                ;DU11 VECTOR ADDRESS DEFINITIONS
                                .ASECT
                                .=      300
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                        .WORD CUR1,340,CUT1,340
                                .CSECT
                                .ENDC
                                        ;
                                        .IF DF  RSX11D
                                        .PSECT  ISR
                                        OFF=60000
                                        .ENDC
                                        .IF DF  RSX11M
000000                                  .PSECT  ISR
                                        .ENDC
                                        .IF NDF RSX11D
000000                                  OFF=0
                                        .ENDC

                                        .IF DF  RSX11D!RSX11M!RT11
                                        .GLOBL  CPVP,DE.PRI
                                        .ENDC
                                        .IF NDF OS
                                        CPVP=   600             ;VECTOR ADDRESS
                                        DE.PRI= 240             ;PRIORITY = 5
                                        .ENDC
```

```
                                        ;DU11 ISR TABLE ADDRESS. ONE IS GIVEN HERE.
                                        ;ADDITIONAL DEVICES MAY BE SUPPORTED BY THE
                                        ;ADDITION OF FUNCTIONALLY SIMILAR TABLES FOR
                                        ;THE ADDITIONAL DEVICES.
                                        ;THE FIRST LABEL OF THE FIRST IS CU.FST,
                                        ;THE FIRST CELL FOLLOWING THE LAST IS CU.LST.
                                              .IF DF RSTS
                                              .GLOBL  CUR1,CUT1,CUS1,CRBLX,CRBUX,CRBINX
                                              .IFF
                                        ;
                                        ;
                                        ;
000000                                  CU.FST:
000000                                  CUR1:
                                              .IF DF  RT11
                                              .GLOBL  DC.PRI
                                              JSP     R5,@RMON        ;NOTIFY RT-11 OF INTERRUPT & CHNG PRIOR.
                                                                      ;ALSO SAVES R5 & R4
                                              .WORD   DC.PRI          ;ARGUMENT FOR INTEN
                                              .ENDC
000000  004167  000652                        JSR R1,CURIS            ;RECEIVE VECTOR ENTRY POINT
000004                                  CUT1:
                                              .IF DF  RT11
                                              JSR     R5,@RMON        ;NOTIFY RT11 OF INTERRUPT & CHNG PRIOR.
                                                                      ;ALSO SAVES R5 & R4
                                              .WORD   DC.PRI          ;ARGUMENT FOR INTEN
                                              .ENDC
000004  004167  000726                        JSR R1,CUTIS            ;TRANSMIT VECTOR ENTRY POINT
                                        ;
                                        ;
                                        ;
000010                                  CUS1:
000010  000000G                               .WORD DUCSR             ;DUCSR ADDRESS FILLS HERE
000012  000000G                               .WORD   CPVP            ;VECTOR ADDRESS
000014     000G                               .BYTE   DE.PRI          ;DEVICE PRIORITY
000015     000                                .BYTE 0                 ;DEVICE RTS COUNTER (TEMP)
000016  000000G                               .WORD C.SPS             ;DEVICE RECEIVE/TRANSMIT BR BUSY LOC
000020  000100'                               .WORD CRBLX+OFF         ;DEVICE CRB LOADER
000022  000160'                               .WORD CRBUX             ;DEVICE CRB UNLOADER
000024  000050'                                WORD CRBINX            ;DEVICE CRB INIT ROUTINE
000026  000000G                               .WORD   CFKLU   ;FAKE SIB LOADER
000030  000000  000000  000000                .WORD 0,0,0             ;DEVICE TRANSMIT TEMPORARY CELLS
000036     000                                .BYTE LNO                       ;LINE NUMBER
000037     000                                .BYTE 0                 ;SYNC IDLE FLAG
000040  000000                                WORD 0                  ;DEVICE PARCSR INITIAL CONTENTS
000042  000000                                .WORD 0                 ;DEVICE DEFERRED STRIP SYNC FLAG
000044  036062                                .WORD   36062           ;DU-11 PARCSR INITIAL CONTENTS
000046  101062                                .WORD   101062          ;DUP-11 PARCSR INITIAL CONTENTS
                                        ;
                                        ;THE FOLLOWING LABEL MUST FOLLOW THE LAST DU11 ISR TABLE.
000050                                  CU.LST:                         ;***** THIS IS THE END OF DU11 ISR TABLES
                                              .ENDC
                                        .IF DF RSTS
                                              DEFORG  RJDV3
                                        .ENDC
```

```
                                ;**********************************
                                ;**********************************
                                ;CIRCULAR RECEIVE BUFFER LOADER AND UNLOADER ROUTINES

                                ;THESE TWO ROUTINES PROVIDE A FACILITY FOR LOADING SINGLE
                                ;WORD ENTRIES INTO AND RETRIEVING SINGLE WORD ENTRIES FROM
                                ;AN INTERNALLY PROVIDED CIRCULAR RECEIVE BUFFER

                                ;BUFFER SIZE IS SPECIFIED AT ASSEMBLY TIME VIA THE PARAMETER CRBSZX

                                ;THE USER MUST DEFINE THE RETURN REGISTER RETNX, AND THE WORK
                                ;REGISTERS CBAX AND CBBX AT ASSEMBLY TIME (CBAX AND CBBX DEFINE WHICH
                                ;REGISTERS THE TAP MUST UTILIZE FOR CRB UNLOADER SERVICE.)

                                ;THE ROUTINES REQUIRE THAT BITS 15 AND 14 BE CLEARED BEFORE THE
                                ;LOADER IS CALLED, SINCE THESE ARE RESERVED FOR USE BY THE ROUTINES

                                ;TO CREATE A UNIQUE COPY OF THESE ROUTINES FOR A MULTI-CRB ENVIRONMENT
                                ;REPLACE X WHEREVER IT APPEARS AT THE END OF A SYMBOL
                                ;BY A UNIQUE CHARACTER
                                ;**********************

                                ;CRB DEFINITIONS (USED BY DU11 LINE 1)

                                .IF NDF RSTS
000400                          CRBSZX=400              ;DEFAULT IS 64 CHARACTER BUFFER
177400                          CRBMZX=-CRBSZX          ;-(THE BUFFER SIZE)
                                .IFF
                                .GLOBL  CRBMZX
                                .ENDC

000005                          CBAX=%5                 ;WORK REGISTER A, MUST BE SAVED
000004                          CBBX=%4                 ;REGISTER B, FOR CHARACTER PASSING
000007                          RETNX=%7                ;RETURN REGISTER

                                ;CIRCULAR RECEIVE BUFFER INITIALIZATION ROUTINE

                                ;MUST BE CALLED PRIOR TO USING CRB LOADER AND UNLOADER

000050  012705  177400  CRBINX: MOV     #CRBMZX,CBAX    ;PICK UP START INDEX
000054  010567  000076          MOV     CBAX,CSINX      ;STORE INDEX
000060  010567  000164          MOV     CBAX,CFINX      ;FETCH INDEX
000064  005065  000652' CRBLPX: CLR     CRBLMX(CBAX)    ;CLEAR CURRENT WORDK
000070  062705  000002          ADD     #2,CBAX         ;DONE?
000074  001373                  BNE     CRBLPX          ;NO. DO NEXT WORD
000076  000207                  RTS     RETNX
```

```
                                        ;CIRCULAR RECEIVE BUFFER LOADER
                                        ;
                                        ;EXIT CONDITIONS:
                                        ;       N=0 - CONTENTS OF CBBX HAVE BEEN STORED IN THE BUFFER
                                        ;       N=1 - BUFFER OVERRUN HAS OCCURED.  CBBX CONTENTS NOT STORED
                                        ;
000100  052704  100000          CRBLX:  BIS     #100000,CBBX    ;SET BIT 15 OF INCOMING WORD
000104  016705  000046                  MOV     CSINX,CBAX      ;FETCH THE CURRENT INDEX
                                        .IF DF  RSX11D
                                        ADD     #OFF,CBAX       ;ADJUST OFFSET FOR MAPPING
                                        .ENDC
000110  005765  000652'                 TST     CRBLMX(CBAX)    ;IS THE LOCATION AVAILABLE?
000114  100413                          BMI     CRBOVX          ;NO.  SET OVERRUN FLAG
000116  010465  000652'                 MOV     CBBX,CRBLMX(CBAX)       ;STORE THE ENTRY
000122  062767  000002  000026          ADD     #2,CSINX        ;UPDATE THE STORE INDEX
000130  001003                          BNE     CRVOX           ;WE'RE NOT AT END OF BUFFER
000132  012767  177400  000016  CRBRSX: MOV     #CRBMZX,CSINX   ;RESET TO START OF BUFFER
000140  000250                  CRVOX:  CLN
000142  000207                          RTS     RETNX
                                        ;
000144  052765  040000  000652' CRBOVX: BIS     #40000,CRBLMX(CBAX)     ;SET BIT 14 OF ENTRY
000152  000270                          SEN
000154  000207                          RTS     RETNX
                                        ;
                                        .IF DF RSTS
                                        .GLOBL  CSINX
                                        .IFF
                                        ;
000156  177400                  CSINX:  .WORD   -CRBSZX          ;CURRENT BUFFER STORE INDEX
                                        .ENDC
```

```
                                ;CIRCULAR RECEIVE BUFFER UNLOADER
                                ;
                                ;EXIT CONTITIONS:
                                ;       N=0,V=0 - CBBX CONTAINS ENTRY.  NO OVERRUN
                                ;       N=0,V=1 - CBBX CONTAINS ENTRY.  OVERRUN HAS OCCURED
                                ;       N=1,V=? - NO ENTRY AVAILABLE
                                ;
000160  016705  000064          CRBUX:  MOV     CFINX,CBAX      ;GET CURRENT FETCH INDEX
000164  016504  000652'                  MOV     CRBLMX(CBAX),CBBX       ;FETCH ENTRY CONTENTS.  WAS IT DATA?
000170  100025                          BPL     CRBNOX          ;NO.  SET N=1 AT EXIT
000172  005165  000652'                  COM     CRBLMX(CBAX)    ;FREE ENTRY LOC (BUT LEAVE DEBUG TRACE)
000176  062767  000002  000044          ADD     #2,CFINX        ;IF AT END OF BUFFER, RESET
000204  001003                          BNE     CRYOX
000206  012767  177400  000034          MOV     #CRBMZX,CFINX
000214  032704  040000          CRYOX:  BIT     #40000,CBBX     ;WAS THERE AN OVERRUN?
000220  001004                          BNE     CRBSVX          ;YES.  SET V
000222  042704  140000                  BIC     #140000,CBBX    ;NO.  CLEAN UP THE ENTRY
000226  000257                          CLALL
000230  000207                          RTS     RETNX
                                ;
000232  042704  140000          CRBSVX: BIC     #140000,CBBX    ;CLEAN UP ENTRY
000236  000250                          CLN
000240  000262                          SEV
000242  000207                          RTS     RETNX
                                ;
000244  000270                  CRBNOX: SEN                     ;NO ENTRY AVAILABLE
000246  000207                          RTS     RETNX
                                ;
                                        .IF DF RSTS
                                        .GLOBL  CFINX,CRBXXX,CRBLMX
                                        .IFF
                                ;
                                ;
000250  177400                  CFINX:  .WORD   -CRBSZX
                                ;
                                ;THE FOLLOWING RESERVES THE BUFFER STOREAGE
                                ;
000252  000652'                  CRBXXX: .=.+CRBSZX
        000652'                         CRBLMX=.                ;BUFFER LIMIT
                                        .ENDC
                                ;*******************
                                ;*******************
                                ;;;;;;;;;;END OF DU11.TBL;;;;;;;;;;
```

```
                                ;
                                ;COMTEX-11 DU11.ISR
                                ;V00-01 JLH 18-OCT-73
                                ;V00-01 JLH 11/15/73
                                ;V02-09 CEP 9/77
                                ;MODIFICATIONS TO RUN WITH CUT CODE
                                ;(SEE DU11.TBL FOR MODIFICATION HISTORY)
                                ;
                                .TITLE DUV209
                                ;
                                ;INSTRUCTION DEFINITIONS AS REQUIRED
000257                          CLALL=  CLC!CLV!CLN!CLZ         ;CLEAR ALL CONDITION CODES
                                ;
                                ;REGISTER DEFINITIONS
177776                          PS=     177776
000000                          R0=     %0
000001                          R1=     %1
000002                          R2=     %2
000003                          R3=     %3
000004                          R4=     %4
000005                          R5=     %5
000006                          SP=     %6
000007                          PC=     %7
                                ;
                                ;BIT DEFINITIONS AS REQUIRED
000001                          BIT00=  1
000002                          BIT01=  2
000004                          BIT02=  4
000010                          BIT03=  10
000020                          BIT04=  20
000040                          BIT05=  40
000100                          BIT06=  100
000200                          BIT07=  200
000400                          BIT08=  400
001000                          BIT09=  1000
002000                          BIT10=  2000
004000                          BIT11=  4000
010000                          BIT12=  10000
020000                          BIT13=  20000
040000                          BIT14=  40000
100000                          BIT15=  100000
                                ;
                                ;DU11 EXTERNALLY REFERENCED GLOBALS
                                .GLOBL C.SPS
                                ;
                                ;DU11 INTERNALLY DEFINED GLOBALS
                                .GLOBL CURIS,CUTIS,CU.OSU,CU.MDM,CU.MOD,CU.INI,CU.KLL,CU.RST
                                .GLOBL CUFSTP,CULSTP
                                ;
                                ;DU11 RXCSR (RECEIVE CONTROL/STATUS REGISTER
000002                          DTR=    BIT01                   ;DATA TERMINAL READY
000004                          RTSA=   BIT02                   ;REQUEST TO SEND STATE
000010                          STD=    BIT03                   ;SECONDARY TRANSMIT DATA
000020                          SSYN=   BIT04                   ;SEARCH SYNC - SET IMPLIES CLEAR RCV ACTIVE
000040                          DSINTE= BIT05                   ;DATA SET INTERRUPT ENABLE
000100                          RINTE=  BIT06                   ;REVEICE INTERRUPT ENABLE
000200                          RDONE=  BIT07                   ;RECEIVE DONE. CHARACTER AVAILABLE
```

```
                        STRSYN= BIT08                   ;STRIP SYNC. CHARACTERS
                        DSR=    BIT09                   ;DATA SET READY
                        SRD=    BIT10                   ;SECONDARY RECEIVE DATA
                        RACT=   BIT11                   ;RECEIVER ACTIVE
                        CD=     BIT12                   ;CARRIER DETECT
                        CTS=    BIT13                   ;CLEAR TO SEND MODEM LEAD
                        RING=   BIT14                   ;RING DETECT MODEM LEAD
                        DSETC=  BIT15                   ;DATA SET CHANGE HAS OCCURRED

                        ;DU11 TXCSR (TRANSMIT CONTROL STATUS REGISTER)
                        HDUPL=  BIT03                   ;HALF DUPLEX STATE
                        SEND=   BIT04                   ;SEND, I.E. TRANSMIT DATA
                        DNAINT= BIT05                   ;DATA NOT AVAILABLE INTERRUPT ENABLE
                        TINTE=  BIT06                   ;TRANSMIT INTERRUPT ENABLE
                        TDONE=  BIT07                   ;TRANSMIT DONE
                        MRESET= BIT08                   ;DEVICE MASTER RESET
                        MMODE=  BIT11+BIT12             ;MAINTENANCE MODE
                        DNA=    BIT15                   ;DATA NOT AVAILABLE
                        ;
                        ;DU11 PARAMETER SELECTION REGISTER PARCSR
                        SYNREG= 377                     ;SYNC CHARACTER BITS 0-7
                        EVNPAR= BIT08                   ;EVEN PARITY SELECTION
                        ENBPAR= BIT09                   ;ENABLE PARITY CHECK/GENERATION
                        CHL68=  BIT10                   ;CHAR LENGTH 6 OR 8 BITS
                        CHL78=  BIT11                   ;CHAR LENGTH 7 OR 8 BITS
                        ;
                        ;DU11 MISCELLANEOUS DEFINITIONS
                        TCOUNT= 2                       ;COUNTER FOR RTS DROP ON TCOMPLETION
                        ;
                        ;DU11 SIB ENTRY TYPES - W/O LINE NUMBER
                        XMTSIB= 1000                    ;TRANSMIT COMPLETE SIB ENTRY - 1ST WORD
                        STSSIB= 40400                   ;STATUS SIB ENTRY - 1ST WORD
                        ;
                        ;DU11 ISR TABLE DEFINTION - RELATIVE TO BEGINNING
                                .IF DF RSTS
                        R=      4
                        S=      10
                                .IFF
                        R=      CUT1-CUR1               ;OFFSET FOR RECEIVE VECTOR ENTRANCE
                        S=      CUS1-CUR1               ;OFFSET FOR TRANSMIT VECTOR ENTRANCE
                                .ENDC
                        RCSR=   S                       ;RECEIVE CSR ADDRESS. WORD
                        RVECT=  RCSR+2                  ;RECEIVE VECTOR ADDRESS. WORD.
                        RPRIOR= RVECT+2                 ;RECEIVE PRIORITY. BYTE
                        TPRIOR= RPRIOR                  ;TRANSMIT PRIORITY. BYTE
                        RTSCTR= TPRIOR+1                ;RTS COUNTER. BYTE
                        RBUSY=  RTSCTR+1                ;RECEIVE BR LEVEL BUSY LOC. WORD
                        TBUSY=  RBUSY                   ;TRANSMIT BR BUSY LOC. WORD
                        CRBLDR= TBUSY+2                 ;CRB LOADER ADDRESS. WORD
                        CRBUNL= CRBLDR+2                ;CRB UNLOADER ADDRESS. WORD
                        CRBINT= CRBUNL+2                ;CRB INITIALIZATION ADDRESS. WORD
                        SIBLDR= CRBINT+2                ;SIB LOADER ADDRESS. WORD
                        NTWP=   SIBLDR+2                ;NEXT TRANSMIT WORD PAIR. WORD
                        CTBA=   NTWP+2                  ;CURRENT TRANSMIT BUFFER ADDRESS
                        CTBC=   CTBA+2                  ;CURRENT TRANSMIT BYTE COUNT. WORD
                        LINENO= CTBC+2                  ;LINE NUMBER. BYTE
                        SYNIDL= LINENO+1                ;SYNC IDLE FLAG. BYTE
```

```
                000040                  SYNCHR= SYNIDL+1                ;SYNC CHARACTER/DEVSTS. WORD
                000040                  DEVSTS= SYNCHR                  ;DEVICE STATUS. HO BYTE OF SYN CHAR. BYTE
                000042                  STPSYN= SYNCHR+2                ;STRIP SYNC FLAG/ DELAYED. BYTE
                000044                  DUSTS=  STPSYN+2                ;DU-11 INITIAL PARCSR SETTING
                000046                  DUPSTS= DUSTS+2                 ;DUP-11 INITIAL PARCSR SETTING
                000050                  ISRSIZ= DUPSTS+2&177776         ;ISR TABLE LENGTH. NOT AN ENTRY.
                                        ;
                                        ;DU11 ISR TABLE POINTERS
000652  000000'                         CUFSTP: .WORD CU.FST            ;ADDRESS OF FIRST DU11 ISR TABLE
000654  000050'                         CULSTP: .WORD CU.LST            ;ADDRESS OF LAST DU11 ISR TABLE
                                        ;
                                        ;2780 EMULATION MODE DEFINITION
                                        .IF NDF RCS
                                        RCS=    1                       ;DEFAULTS TO 2780 EMULATION USE
                                        .ENDC
                                        ;
                                        ;VECTOR DEFINTION. DEFAULTS TO UTILIZE .ASECT CODING
                                        .IF NDF VECTOR
                                        VECTOR= 0                       ;DEFAULT TO COMMON COMTEX PRACTICE
                                        .ENDC
                                        ;
                                        ;PARITY CALCULATION DEFINITION
                                        .IF NDF USEPAR
                                        USEPAR= 0                       ;DEFAULT TO ODD/EVEN MEANING
                                        .ENDC
```

```
                                        ;DU11 RECEIVE INTERRUPT SERVICE ROUTINE -- CURIS
                                        ;ENTERED FROM ISR TABLE VIA JSR R1,CURIS.
                                        ;UPON ENTRY ROUTINE MUST SET LEVEL BUSY,
                                        ;LOWER PRIORITY, AND SAVE ANY NECCESSARY REGISTERS.
                                        ;IT MUST EXIT VIA JMP C.SPS (PRIORITY SCHEDULER)
                                        ;AFTER RESTORING ANY REGISTERS AND CLEARING THE
                                        ;LEVEL BUSY FLAG.
                                        ;
000656                                  CURIS:
                                                .IF NDF RSTS    ;RSTS HAS SAVED ALL REGS
000656  010246                                  MOV R2,-(SP)            ;THEN SAVE REGISTERS R2-R5 FOR USE
000660  010346                                  MOV R3,-(SP)
                                                .IF NDF RSX11M&RT11     ;$INTSV ALREADY HAS SAVED R4 & R5
                                                MOV R4,-(SP)
                                                MOV R5,-(SP)
                                                .ENDC
                                                .ENDC
000662  016102  000004                          MOV RCSR-R(R1),R2       ;GET DEVICE RCSR ADDRESS
                                        ;
                                        ;RCV INTERRUPT OCCURRED.  TEST FOR INTERRUPT CAUSE.
000666  011203                          RGOT:   MOV @R2,R3              ;OBTAIN RCSR CONTENTS
                                        ;
                                        ;INTERRUPT CAUSE WAS NOT DATA SET CHANGE.  DID RCV CHAR OCCUR.
000670  105703                          CUTRDN: TSTB R3                 ;IS RCV DONE SET
000672  100015                                  BPL CURXIT              ;NO, NO SERVICE, LEAVE
000674  046112  000036                          BIC STPSYN-R(R1),@R2    ;CLEAR STRIP SYNC IF DEFERRED FLAG SET
                                        ;
                                        ;RECEIVE CHARACTER INTERRUPT HAS OCCURRED.  SERVICE CHARACTER
000700  016204  000002                  CUGRCH: MOV 2(R2),R4            ;LOAD RECIEVED CHARACTER INTO R4
000704  010405                                  MOV R4,R5               ;AND REPLICATE CHARACTER
000706  030527  040000                          BIT R5,#BIT14           ;DID OVERRUN OCCR
000712  001402                                  BEQ NORCV0              ;NO
000714  052704  020000                          BIS #BIT13,R4           ;YES,SET INDICATOR
000720  004771  000014                  NORCV0: JSR PC,@CRBLDR-R(R1)    ;STORE CHARACTER IN CRB
000724  000760                                  BR RGOT                 ;AND LOOK FOR MORE
                                        ;
                                        ;
                                        ;RECEIVE INTERRUPT ROUTINE EXIT POINT
000726                                  CURXIT:
                                                .IF NDF RSTS
                                                .IF NDF RSX11M&RT11
                                                MOV (SP)+,R5            ;RESTORE REGISTERS R5-R2
                                                MOV (SP)+,R4
                                                .ENDC
000726  012603                                  MOV (SP)+,R3
000730  012602                                  MOV (SP)+,R2
                                                .ENDC
000732  012601                                  MOV (SP)+,R1            ;AND FINALLY RESTORE ENTRY REGISTRE
                                                .IF DF  RSTS
                                                RETURN
                                                .ENDC
                                                .IF DF  RSX11D
                                                JMP     @#..INTX
                                                .ENDC
                                                .IF DF  RSX11M!RT11
000734  000207                                  RTS     PC              ;RETURN TO CODE IN THE EXEC BUFFER POOL
                                                .ENDC
```

```
                                        ;TRANSMIT INTERRUPT SERVICE ROUTINE - CUTIS
                                        ;ENTERED FROM ISR TABLE VIA JSR R1,CUTIS
                                        ;UPON ENTRY ROUTINE MUST SET LEVEL BUSY
                                        ;LOWER PRIORITY LEVEL TO THAT OF DEVICE, AND SAVE
                                        ;ANY NECCESSARY REGISTERS.
                                        ;UPON EXIT, IT MUST RESTORE REGISTERS, CLEAR
                                        ;LEVEL BUSY AND EXIT VIA JMP C.SPS (PRIORITY SCHEDULER).
                                        ;
000736                                  CUTIS:
                                                .IF NDF RSTS            ;RSTS ALREADY SAVED ALL REGS
000736  010246                                  MOV R2,-(SP)            ;SAVE REGISTERS R2,R3
000740  010346                                  MOV R3,-(SP)
000742  010046                                  MOV     R0,-(SP)        ;RSX11D NEEDS R0 ALSO
                                                .ENDC
000744  016102  000000                          MOV RCSR-S(R1),R2       ;OBTAIN RCSR ADDRESS
                                        ;
                                        ;INTERRUPT HAS OCCURED. GET TCSR CONTENTS.
000750  016203  000004                          MOV 4(R2),R3            ;OBTAIN TCSR CONTENTS
000754  105703                                  TSTB R3                 ;IS TDONE SET
000756  100104                                  BPL CUTEXT              ;NOT NOW, EXIT FROM INT SVC RTN
000760  105761  000005                          TSTB RTSCTR-S(R1)       ;IS RTS COUNTER ACTIVE
000764  001406                                  BEQ TRANSI              ;NO - TRY TRANSMISSION
000766  100071                                  BPL WINDUP              ;WAITING FOR RTS DROP
                                        ;
                                        ;WAITING FOR CTS PRESENCE BEFORE TRANSMISSION
000770  031227  020000                          BIT @R2,#CTS            ;IS CTS PRESENT
000774  001450                                  BEQ OUTCHR              ;NO, SEND CHARACTERS TO WAIT
000776  105061  000005                          CLRB RTSCTR-S(R1)       ;CTS PRESENT, CLEAR DELAY FLAG
                                        ;
                                        ;NOW TRY TO SEND SOME DATA CHARACTERS
001002  005761  000024                  TRANSI: TST CTBC-S(R1)          ;IS TRANSMISSION ACTIVE
001006  001456                                  BEQ EXITT               ;TRANSMISSION NOT ACTIVE, CLEAR INTERRUPTS
                                        ;
                                        ;TRANSMISSION IS IN PROGRESS. OUTPUT DATA CHARACTER
001010  117162  000022  000006          TRANS:  MOVB @CTBA-S(R1),6(R2)  ;OBTAIN TRANSMIT CHARACTER
001016  005261  000022                          INC CTBA-S(R1)          ;UPDATE ADDRESS
001022  005361  000024                          DEC CTBC-S(R1)          ;AND BYTE COUNT REMAINING
001026  001060                                  BNE CUTEXT              ;TRANSMIT DEFINITELY NOT COMPLETE
                                        ;
                                        ;TRANSMIT BYTE COUNT WENT TO ZERO. UPDATE TO NEXT CA,BC PAIR
                                                .IF NDF RSX11M
                                                MOV     R5,-(SP)        ;SAVE R5
                                                .ENDC
001030  016105  000020                          MOV NTWP-S(R1),R5       ;AND OBTAIN NEXT TWP - BA SPECIFICALLY
001034  012561  000022                          MOV (R5)+,CTBA-S(R1)    ;OBTAIN NEXT BUFFER ADDRESS
001040  001405                                  BEQ TFINI               ;THAT'S END OF THIS TRANSMISSION
                                                .IF DF  RSX11D
                                                ADD     #OFF,CTBA-S(R1)
                                                .ENDC
001042  012561  000024                          MOV (R5)+,CTBC-S(R1)    ;THEN UPDATE BYTE COUNT
001046  010561  000020                          MOV R5,NTWP-S(R1)       ;AND FINALLY NEXT T WORD PAIR
001052                                  CUTXT1: .IF NDF RSX11M
                                                MOV     (SP)+,R5        ;RESTORE R5
                                                .ENDC
001052  000446                                  BR CUTEXT               ;AND EXIT FROM INTERRUPT
                                        ;
                                        ;TRANSMIT BUFFER ADDRESS HAS GONE TO ZERO.  TERMINATE TRANSMISSION.
```

```
                                TFINI:
                                        .IF DF RSX
001054  016700  000000G                 MOV     TASKAD,R0       ;
                                        .IF DF RSX11D
                                        BIS     #40,A.EF(R0)            ;SET 'SIB EVENT FALG'
                                        BISB    #EV.SE,@#.SERFG         ;SET SIGNIFICANT EVENT
                                        .IFF
001060  052760  000040  000000G         BIS     #40,T.EFLG(R0)
001066  016767  000000G 000000G         MOV     $TSKHD,$RQSCH   ;DECLARE SIG EVENT
001074  005067  000000G                 CLR     $SIGFL
                                        .ENDC
                                        .IFF
                                        MOV     #XMTSIB,R5      ;SET UP SIB ENTRY FOR TAP NOTIFICATION
                                        CLRB    R5              ;
                                        JSR PC,@SIBLDR-S(R1)    ;AND CALL SIB INSERTION ROUTINE
                                        .ENDC
                                        .IF NDF RSX11M
                                        MOV     (SP)+,R5        ;RESTORE R5
                                        .ENDC
                                ;
                                ;TAP NOTIFICATION HAS OCCURRED.  NOW CLEAR INTERRUPTS OR USE DELAY
001100  105761  000027                  TSTB SYNIDL-S(R1)       ;IF SYNIDL SET, USE SYNCS FOR DELAY
001104  001017                          BNE EXITT               ;CLEAR TRANSMIT INTERRUPTS
001106  112761  000002  000005          MOVB #TCOUNT,RTSCTR-S(R1) ;OBTAIN RTS DROP COUNTER
001114  000425                          BR CUTEXT               ;AND EXIT INTERRUPT ROUTINE
                                ;
                                ;WAIT IN PROGRESS HAS NOT COMPLETED SEND SOME DATA
001116  105761  000027          OUTCHR: TSTB SYNIDL-S(R1)       ;IS LINE IDLING
001122  001004                          BNE OUTSYN              ;YES, LET IT SLEEP PEACEFULLY
                                ;
                                ;LINE IS DELAYING - REQUIRES MARK CHARACTER FOR DELAYING
001124  012762  000777  000006  OUTMRK: MOV #777,6(R2)           ;SEND THIS CHARACTER
001132  000416                          BR CUTEXT               ;AND TEST FOR ADDITIONAL OUTPUT REQUIREMENT
                                ;
                                ;LINE IS DELAYING - REQUIRES SYNC CHARACTER
001134  116162  000030  000006  OUTSYN: MOVB SYNCHR-S(R1),6(R2) ;OUTPUT SYN CHARACTER
001142  000412                          BR CUTEXT               ;AND LOOP UNTIL NO TRANSMIT FLAG
                                ;
                                ;LINE HAS COMPLETED - OR FALSE INTERRUPT
001144  042703  000140          EXITT:  BIC #TINTE+DNAINT,R3    ;STOP THE WORLD
001150  000407                          BR CUTEXT               ;AND EXIT INTERRUPT ROUTINE
                                ;
                                ;WAIT IS IN PROGRESS FOR TRANSMISSION TERMINATION
001152  105361  000005          WINDUP: DECB RTSCTR-S(R1)       ;COUNT THESE INTERRUPTS
001156  001362                          BNE OUTMRK              ;PROCESS NOT COMPLETE, WAIT BY SENDING CHAR
                                ;
                                ;TERMINATION PROCESS COMPLETE.  TELL THE WORLD
001160  042712  000004                  BIC #RTSA,@R2           ;NO, CLEAR RTS ASSERTION
001164  042703  000020                  BIC #SEND,R3            ;AND DROP SEND REQUEST
                                ;
                                ;EXIT FROM TRANSMIT INTERRUPT SERVICE
001170  010362  000004          CUTEXT: MOV R3,4(R2)            ;AND UPDATE TCSR PRIOR TO EXIT
                                        .IF NDF RSTS
001174  012600                          MOV     (SP)+,R0        ;
001176  012603                          MOV (SP)+,R3            ;THEN RESTORE R3 AND R2
001200  012602                          MOV (SP)+,R2
                                        .ENDC
```

```
    313 001202  012601                          MOV (SP)+,R1            ;FINALLY RESTORE ENTRY REGISTER
    314                                         .IF DF  RSX11D
    315                                         JMP     @#..INTX
    316                                         .ENDC
    317                                         .IF DF  RSTS
    318                                         RETURN
    319                                         .ENDC
    320                                         .IF DF  RSX11M!RT11
    321 001204  000207                          RTS     PC              ;RETURN TO CODE IN THE EXEC BUFFER POOL
    322                                         .ENDC
```

```
                                ;DU11 ISR NON-INTERRUPT SERVICE ROUTINES
                                ;ENTERED FROM COMTEX-11 SCIP DEVICE TYPE MATRIX SWITCH (C. SARP)
                                ;WITH THE FOLLOWING CONDITIONS ON ENTRY:
                                ;R1=POINTER TO PRIORITY LIST
                                ;R2=TAP LINE TABLE ADDRESS
                                ;R3=LINE NUMBER AS WORD ADDRESS
                                ;EXIT CONDITIONS. EXITS VIA RTS PC
                                ;SP=STATE UPON ENTRY
                                ;ALL OTHER REGISTERS UNDEFINED
                                ;PS RETURNED TO CALLER
                                ;
                                ;DU11 TRANSMIT ENABLE ROUTINE - CU.OSU
                                ;DEVICE TYPE MATRIX SWITCH FUNCTION CODE = 0
                                ;
                                        .IF DF  RSX11D!RSX11M
000000                                  .PSECT  XXX
                                        .ENDC
                                .IF NZ RCS
                                .GLOBL CP.OSU
000000                          CP.OSU:
                                .ENDC
                                ;
000000  016202  000002          CU.OSU: MOV 2(R2),R2            ;OBTAIN ISR TABLE ADDRESS
                                ;
                                ;END OF TRANSMIT CHAIN HAS BEEN REACHED WITH NO ERROR
000004  012105                  CUOSU1: MOV (R1)+,R5            ;OBTAIN FLAG WORD AS SPECIFIED
                                ;
                                ;NO ERROR IN WORD/BYTE SPECIFICATION, TRY IDLE SYNC CHANGE
000006  016203  000010          CUOSU2: MOV RCSR(R2),R3         ;GET DEVICE RCSR ADDRESS
                                        .IF DF LSI11
                                        MFPS    -(SP)           ;THEN SAVE CURRENT PRIORITY
                                        MTPS    TPRIOR(R2)      ;AND DISALLOW INTERRUPTS FROM THIS DEVICE
                                        .IFF
000012  113746  177776                  MOVB @#PS,-(SP)         ;THEN SAVE CURRENT PRIORITY
000016  116237  000014  177776          MOVB TPRIOR(R2),@#PS    ;AND DISALLOW INTERRUPTS FROM THIS DEVICE
                                        .ENDC
000024  005705                         TST R5                  ;;;IS CHANGE OF IDLE SYNC SPECD
000026  100006                         BPL CUOSU3              ;;;NO
000030  105062  000037                 CLRB SYNIDL(R2)         ;;;YES, CLEAR IDLE SYNC FLAG
000034  006305                         ASL R5                  ;;; AND SEE WHAT IS TO BE DONE
000036  100002                         BPL CUOSU3              ;;;CLEAR SYNC IDLE
000040  105262  000037                 INCB SYNIDL(R2)         ;;;ASSERT SYN IDLE
                                ;
                                ;FLAG TESTING HAS BEEN COMPLETED.  NOW START UP WHATEVER IS TO BE
000044  012104                  CUOSU3: MOV (R1)+,R4            ;;;CHARACTERS TO BE TRANSMITTED
000046  001423                         BEQ CUOSU4              ;;;NO
000050  010462  000032                 MOV R4,CTBA(R2)         ;;;YES, UPDATE BUFFER ADDRESS
                                        .IF DF  RSX11D
                                        ADD     #OFF,CTBA(R2)
                                        .ENDC
000054  012162  000034                 MOV (R1)+,CTBC(R2)      ;;;AND THEN BYTE COUNT
                                        .IF DF  RSX11D
                                        ADD     #OFF,R1
                                        .ENDC
000060  010162  000030                 MOV R1,NTWP(R2)         ;;;AND NEXT TRANSMIT WORD PAIR
000064  052713  000004                 BIS #RTSA,@R3           ;;;ASSERT RTS
000070  052763  000120  000004         BIS #SEND+TINTE,4(R3)   ;;;AND ASSERT SEND WITH TDNA INT
```

```
                                                BIT #MMODE,4(R3)        ;;;IS MAINTENANCE MODE SET
                                                BNE CUTSTX              ;;;IF SO IGNORE WAIT FOR CTS
                                                MOVB #-1,RTSCTR(P2)     ;;;THEN INDICATE CTS WAIT
                                                BR CUTSTX               ;;; THEN START UP DEVICE IF POSSIBLE
                                        ;
                                        ;TRANSMISSION, IF STARTABLE HAS BEEN INITED
                                        CUOSU4: TST CTBA(R2)            ;;;IS TRANSMISSION ACTIVE
                                                BNE CUOSUX              ;;;YES, NO CAHNGES NECCESSARY
                                                TSTB SYNIDL(R2)         ;;;IS SYN IDLE ACTIVD
                                                BEQ CUOSU5              ;;;NO
                                        ;
                                        ;SYNC IDLE IS TO BE ASSERTED
                                                BIS #SEND,4(R3)         ;;;ASSERT SEND
                                                BIS #RTSA,@R3           ;;;AND ASSERT REQUEST TO SEND
                                        CUTSTX: TSTB 4(R3)              ;;;TEST IF TDONE SET
                                                BPL CUOSUX              ;;;NO, LEAVE. DEVICE SHOULD START
                                                TSTB SYNIDL(R2)         ;;;IS SYNC IDLE ACTIVE
                                                BNE CUOSU6              ;;;YES, SEND ONE SYN CHARACTER
                                        ;
                                        ;START TRANSMISSION VIA JAMMING MARKS INTO T REGISTER
                                                MOV #777,6(R3)          ;;;8 BIT MARK CHARACTER PLUS TSOM BIT FOR DUP
                                                BR CUOSUX               ;;;AND EXIT
                                        ;
                                        ;START TRANSMISSION VIA SYNC CHARACTER OUTPUT
                                        CUOSU6: MOVB SYNCHR(R2),6(R3)   ;;;LOAD T REGISTER WITH SYNC TO START XMISSION
                                                BR CUOSUX               ;;;THEN EXIT
                                        ;
                                        ;SYNC IDLE IS TO BE CLEARED
                                        CUOSU5: BIC #RTSA,@R3           ;;;DROP RTS ASSERTION
                                                BIC #SEND,4(R3)         ;;;AND SEND ASSERTION
                                        ;
                                        ;ALL PROCESSING COMPLETE. EXIT
                                        CUOSUX: .IF DF LSI11
                                                MTPS    (SP)+           ;DROP TO ENTRANCE PRIORITY LEVEL
                                                .IFF
                                                MOVB (SP)+,@#PS         ;DROP TO ENTRANCE PRIORITY LEVEL
                                                .ENDC
                                                CLALL                   ;AND CLEAR ALL CONDITION CODES
                                                RTS PC                  ;THEN EXIT
```

```
                                        ;DU11 MODEM CONTROL SENSE ROUTINE - CU.MDM
                                        ;ENTERED FROM SCIP DEVICE TYPE MATRIX SWITCH
                                        ;DEVICE TYPE MATRIX SWITCH FUNCTION CODE = 1
                                        ;
                                        .IF NZ RCS
                                        .GLOBL CP.MDM
000222                                  CP.MDM:
                                        .ENDC
                                        ;
000222  016202  000002                  CU.MDM: MOV 2(R2),R2            ;OBTAIN ISR TABLE ADDRESS
000226  012746  177777                          MOV #-1,-(SP)           ;SET FLAG TO EXIT WUTH ERROR
000232  016104  000002                          MOV 2(R1),R4            ;OBTAIN SPECIFIED PARAMETER WORD 1
000236  030427  100121                          BIT R4,#BIT00+BIT04+BIT06+BIT15 ;IS ANY ACTION SPECIFIED
000242  001433                              BEQ NOSRDP                  ;NOTHING SPECIFIED, EXIT WITH ERROR
000244  016203  000010                      MOV RCSR(R2),R3             ;SOMETHING TO DO.  GET RCSR ADDRESS
000250  005016                              CLR @SP                     ;AND SET ERROR FLAG TO NO ERROR
                                        ;
                                        ;SOME ACTION IS SPECIFIED.  TEST EACH CONDITION
000252  030427  000001                      BIT R4,#BIT00               ;IS DTR CHANGE REQUESTED
000256  001405                              BEQ NODTRC                  ;NO DTR CHANGE
000260  030427  000002                      BIT R4,#BIT01               ;DTR CHANGE SPECIFIED, HOW?
000264  001424                              BEQ CLRDTR                  ;CHANGE BY CLEARING
000266  052713  000002                      BIS #DTR,@R3                ;CHANGE BY ASSERTION OF DTR
000272                                  NODTRC:
000272  005704                          NOSTDC: TST R4                  ;IS SENSE REQUESTED
000274  100016                              BPL NOSRDP                  ;NO, EXIT
000276  005004                              CLR R4                      ;YES, CLEAR SENSE REPOSITORY
000300  011303                              MOV @R3,R3                  ;OBTAIN CURRENT STATES
000302  030327  001000                  NOCTSP: BIT R3,#DSR             ;IS DSR PRESENT
000306  001402                              BEQ NODSRP                  ;NO
000310  052704  000004                      BIS #BIT02,R4               ;YES, INDICATE DSR PRESENT
000314  030327  000002                  NODSRP: BIT R3,#DTR             ;IS DTR ASSERTED
000320  001402                              BEQ NODTRP                  ;NO
000322  052704  000010                      BIS #BIT03,R4               ;YES, INDICATE DTR ASSERTION
000326                                  NODTRP:
000326  010461  000004                  NSRDPX: MOV R4,4(R1)            ;AND UPDATE USER SENSE LNCS
                                        ;
                                        ;THEN EXIT FROM MODEM CONTROL ROUTINE.
000332  005726                          NOSRDP: TST (SP)+               ;OBTAIN ERROR FLAGS, IF ANY
000334  000207                              RTS PC                      ;AND EXIT
                                        ;
                                        ;ROUTINE IS REQUESTED TO CLEAR DTR
000336  042713  000002                  CLRDTR: BIC #DTR,@R3            ;CLEAR DTR ASSERTION
000342  000753                              BR NODTRC                   ;AND CONTINUE TESTS
                                        ;
                                        ;ROUTINE IS REQUESTED TO CLEAR STD
000344  042713  000010                  CLRSTD: BIC #STD,@R3            ;CLEAR STD ASSERTION
000350  000750                              BR NOSTDC                   ;THEN CONTINUE TESTS
```

```
                                        ;DU11 MODE CONTROL/SENSE ROUTINE - CU.MOD
                                        ;DEVICE TYPE MATRIX SWITCH FUNCTION CODE = 2
                                                .IF DF RSX
001206                                          .PSECT ISR
                                                .ENDC
                                        ;
                                        .GLOBL CP.MOD
001206                                  CP.MOD:
                                        ;
001206  016202  000002                  CU.MOD: MOV 2(R2),R2            ;OBTAIN ISR TABLE ADDRESS
001212  016104  000002                          MOV 2(R1),R4            ;OBTAIN PARAMETER WORD
001216  012746  177777                          MOV #-1,-(SP)           ;INITIALIZE ERROR FLAG TO ERROR PRESENT
001222  016203  000010                          MOV RCSR(R2),R3         ;OBTAIN RCSR ADDRESS
001226  016205  000040                          MOV SYNCHR(R2),R5       ;OBTAIN DEVICE SYN CHAR/ STATUS WORD
001232  030427  015041                          BIT R4,#BIT00+BIT05+BIT09+BIT11+BIT12 ;IS ANYTHIG TO BE DONE
001236  001460                                  BEQ TSTSNS              ;NO CHANGES, TRY SENSE REQUEST
001240  005016                                  CLR @SP                 ;SOMETHING SPECIFIED, CLEAR ERROR FLAG
001242  030427  000001                          BIT R4,#BIT00           ;ARE INT/MMODE SPECS TO BE CHANGED
001246  001405                                  BEQ NOINTC              ;NO
001250  030427  000010          CKRCVI: BIT R4,#BIT03                   ;IS RECEIVE ENABLE TO BE ASSERTED
001254  001440                                  BEQ CLRRI               ;CLEAR
001256  052713  000120                          BIS #RINTE+SSYN,@R3     ;ENABLE RCV INTERRUPT AND SEARCH SYNC
001262                                  CKSTSI:
001262                                  NOINTC:
001262  052705  006000                          BIS     #CHL78!CHL68,R5 ;SET UP FOR 8 BITS
001266                                  NOCLC:
001266  052763  000010  000004                  BIS #HDUPL,4(R3)        ;ASSERT HALF DUPLEX MODE
001274                                  NOHDC:
001274  030427  010000                  NOSYNC: BIT R4,#BIT12           ;IS STRIP SYNC CONTROL TO BE CAHNGED
001300  001413                                  BEQ NOSSC               ;NO CHANGE
001302  030427  020000                          BIT R4,#BIT13           ;CHANGE REQUESTED, HOW IS IT TO BE SET
001306  001426                                  BEQ CLRSS               ;CLEAR STRIP SYNC
001310  052713  000400                          BIS #STRSYN,@R3         ;ASSERT STRIP SYNC
001314  030427  040000          CKSSDC: BIT R4,#BIT14                   ;IS STRIP SYNC DEFERRED SELECTION ON
001320  001424                                  BEQ CLRSSD              ;CLEAR DEFFERRED STRIP SYNC SELECTION
001322  012762  000400  000042                  MOV #STRSYN,STPSYN(R2)  ;SET DEFERRED STRIP SYNC CONTROL
001330  005704                          NOSSC:  TST R4                  ;IS RCV ACTIVE TO BE CLEARED
001332  100004                                  BPL NORCVC              ;NO
001334  042713  000020                          BIC #SSYN,@R3           ;CLEAR RCV ACTIVE VIA SEARCH SYN CLEAR
001340  052713  000020                          BIS #SSYN,@R3           ;AND SEARCH SYNC ASSERTION
001344  010563  000002          NORCVC: MOV R5,2(R3)                    ;THEN UPDATE DEVICE STATUS
001350  010562  000040                          MOV R5,SYNCHR(R2)       ;AND SAVED DEVICE STATUS
001354  000411                                  BR TSTSNS               ;AND THEN TEST FOR SENSING REQUIRED
                                        ;
                                        ;CLEAR RECEIVE INTERRUPT SELECTION - DISABLE RECEIVE
001356  042713  000120          CLRRI:  BIC #RINTE+SSYN,@R3             ;CLEAR RINT AND SEARCH SYNC
001362  000737                          BR CKSTSI                       ;AND CONTINUE TESTS
                                        ;
                                        ;CLEAR STRIP SYNC SELECTION
001364  042713  000400          CLRSS:  BIC #STRSYN,@R3 ;DROP STRIP SYNC SELECTION
001370  000751                          BR CKSSDC
                                        ;
                                        ;CLEAR DEFERRED STRIP SYNC SELECTION
001372  005062  000042          CLRSSD: CLR STPSYN(R2)                  ;DROP ISR TABLE INDICATOR
001376  000754                          BR NOSSC                        ;AND CONTINUE
                                        ;
001400  005726                  TSTSNS: MODYIT:   TST (SP)+             ;RECOVER FLAG WORD
```

```
    527 001402  000207                        RTS PC                  AND EXIT
    528                                       .IF DF RSX
    529 000352                                .PSECT XXX
    530                                       .ENDC
```

```
                                ;DU11 LINE INITIALIZATION - CU.INI
                                ;MATRIX SWITCH FUNCTION CODE = 3
                                ;
                                .IF NZ RCS
                                .GLOBL CP.INI
000352                          CP.INI:
                                .ENDC
                                ;
000352                          CU.INI:
000352  016700  000652'                MOV CUFSTP,R0           ;OBTAIN ADDRESS OF BEGINNING LINE TABLE
                                ,
000356  000405                          BR      GOTLIN
                                ;
                                ;
000360  004567  000000G         ERROR1: JSR     R5,C.SRES       ;MAKE SURE WE RESTORE REGS ON ERROR
000364                          CU.IN1:
000364  000257                  ERROR:  CLALL
000366  000270                          SEN
000370  000207                          RTS PC                  ;EXIT ROUTINE
                                ;
                                ;ISR TABLE FOR SPECIFIED LINE FOUND
000372                          GOTLIN: .IF DF RSTS
                                        CMP     RCSR(R0),#160000        ;IS DEVICE ADDRESS PRESENT?
                                        BHI     DEVPRS                  ;YES
                                        MOV     @RCSR(R0),R3    ;GET ADDRESS RSTS IS GIVING US
                                        MOV     R3,RCSR(R0)     ;PUT CSR ADD INTO THE FIELD - CAUSE THAT
                                                                ;IS WHAT THE REST OF THE CODE EXPECTS TO FIND
                                        .ENDC
000372                          DEVPRS:
000372  016003  000010                  MOV     RCSR(R0),R3     ;GET RCSR ADDRESS OF SPECIFIED DU11
000376  020327  160000                  CMP     R3,#160000      ;WAS IT FOUND
000402  103770                          BLO     ERROR           ;NO - THERE IS A CONFIG ERRER
                                        .IF DF RSX11D!RSX11M!RT11
000404  004567  000000G                 JSR     R5,C.SREG               ;DO THIS HERE SO THAT WE DON'T
                                                                        ;CLOBBER THE CSR IF NETWORKS IS
                                                                        ;USING THE DEVICE
000410  010005                          MOV     R0,R5           ;COPY TABLE ADDRESS
000412  016500  000012                  MOV     RVECT(R5),R0    ;GET VECTOR ADD IN R0
000416  016503  000014                  MOV     RPRIOR(R5),R3           ;SET DEV INT PRIORITY IN R3
000422  005002                          CLR     R2                      ;
000424  012701  000000'                 MOV     #CUR1,R1                ;SET UP FOR RECEIVE INT
000430  004737  000000G                 JSR     PC,@#..CINT             ;CALL ROUTINE TO SET UP VECTOR
000434  103751                          BCS     ERROR1                  ;VECTOR ALREADY ASSIGNED
000436  022020                          CMP     (R0)+,(R0)+             ;GET TRAN VECTOR
000440  012701  000004'                 MOV     #CUT1,R1                ;
000444  004737  000000G                 JSR     PC,@#..CINT             ;GO SET UP TRANSMIT VECTOR
000450  103743                          BCS     ERROR1                  ;VECTOR ALREADY ASSIGNED
000452  004567  000000G                 JSR     R5,C.SRES               ;RESTORE REGISTERS
                                        .ENDC
000456  010062  000002                  MOV R0,2(R2)            ; PLACE ISR TABLE ADDRESS IN TAP TABLE
000462  016060  000044  000040          MOV     DUSTS(R0),SYNCHR(R0)    ;ASSUME WE HAVE DU-11
000470  052763  000001  000004          BIS     #BIT00,4(R3)    ;FIND OUT IF IS DU (DUV) OR DUP
000476  032763  000001  000004          BIT     #BIT00,4(R3)    ;DID THE BIT REFUSE TO SET?
000504  001003                          BNE     10$             ;NO, IT IS A DU-11
000506  016060  000046  000040          MOV     DUPSTS(R0),SYNCHR(R0)   ;YES, ITS A DUP-11
000514  052763  000400  000004  10$:    BIS #MRESET,4(R3)       ;THEN CLEAR DEVICE
000522  016063  000040  000002          MOV SYNCHR(R0),2(R3)    ;AND SET DEVICE TO OLD STATES
```

```
                                                JSR PC,@CRBINT(R0)      ;THEN CALL CRB INIT RTN
                                                MOV CRBUNL(R0),4(R2)    ;PLACE CRB UNLOADER ADDRESS IN TAP TABLE
                                                CLR NTWP(R0)            ;THEN INITIALIZE ISR TABLE
                                                CLR CTBA(R0)
                                                CLR CTBC(R0)
                                                CLRB SYNIDL(R0)
                                                CLRB RTSCTR(R0)
                                                CLR STPSYN(R0)
                                                .IF DF  RT11
                                                MOV     #AT.END,R0      ;SET UP FOR EXIT TIME
                                                .DEVICE                 ;DISABLE INTERRUPT WHEN THE
                                                                        ;PROGRAM ABORTS FOR ANY REASON
                                                .ENDC
                                                CLALL                   ;CLEAR ERROR FLAGS
                                                RTS PC                  ;EXIT

                                                .IF DF RT11
                                        ;;;;;;;;;;;;;;;;;;;;
                                        ; PARAMETER BLOCK FOR .DEVICE CALL
                                        AT.END: .BYTE   0               ;NO CHANNEL
                                                .BYTE   14              ;FUNCTION CODE
                                                .WORD   ATTEND          ;LIST OF WORDS TO BE LOADED
                                                                        ;AT PROGRAM TERMINATION
                                        ;
                                        ;
                                        ATTEND: .WORD   DUCSR           ;ADDRESS OF COMM CSW
                                                .WORD   0               ;FORCE IT TO 0
                                                .WORD   DUCSR+4         ;TRANSMIT CSW
                                                .WORD   0               ;ALSO FORCE TO 0
                                                .WORD   0               ;END OF LIST
                                        ;
                                        ;;;;;;;;;;;;;;;;;;
                                                .ENDC
```

```
                                ;DU11 SYSTEM RESTART - CU.RST
                                ;MATRIX SWITCH FUNCTION CODE = 4
                                ;
                                .IF NZ RCS
                                .GLOBL CP.RST
000576                          CP.RST:
                                .ENDC
                                ;
000576                          CU.RST:
                                        .IF NDF RSTS
000576  016705  000652'                 MOV     C!JFSTP,R5      ;GET BEGINNING TABLE ADDRESS
                                        .IF DF RSX11D!RSX11M!RT11
000602  016500  000012                 MOV RVECT(R5),R0        ;OBTAIN VECTOR ADDRESS
000606  004737  000000G                JSR PC,@#..DINT         ;AND DISCONNECT FROM VECTOR
000612  022020                         CMP (R0)+,(R0)+         ;BUMP TO TRANSMIT VECTOR
000614  004737  000000G                JSR PC,@#..DINT         ;AND DISCONNECT FROM THAT VECTOR
                                        .ENDC
000620  016503  000010                 MOV     RCSR(R5),R3     ;AND OBTAIN RCSR ADDRESS
000624  052763  000400  000004         BIS     #MRESET,4(R3)   ;CLEAR THE DEVICE
                                        .ENDC
000632  000257                         CLALL           ;CLEAR CONDITION CODES
000634  000207                         RTS PC                  ;AND RETURN TO CALLER
```

```
                                ;DU11 TRANSMIT DISABLE ROUTINE - CU.KLL
                                ;DEVICE TYPE MATRIX SWITCH FUNCTION CODE = 5
                                ;
                                .IF NZ RCS
                                .GLOBL CP.KLL
000636                          CP.KLL:
                                .ENDC
                                ;
000636  000257                 CU.KLL: CLALL                   ;NO, ERROR EXISTS
000640  000270                         SEN
000642  000207                         RTS PC                  ;AND EXIT
                                ;
000001                                  .END
```

```
DUV209  MACRO M1104  06-DEC-77 08:20  PAGE 18-1
SYMBOL TABLE

BIT00 = 000001          CP.MDM  000222RG    003 CUTIS   000736RG    002 MMODE = 014000          RTSCTR= 000015
BIT01 = 000002          CP.MOD  001206RG    002 CUTRDN  000670R     002 MODXIT  001400R     002 RVECT = 000012
BIT02 = 000004          CP.OSU  000000RG    003 CUTSTX  000144R     003 MRESET= 000400          S     = 000010
BIT03 = 000010          CP.RST  000576RG    003 CUTXT1  001052R     002 NOCLC   001266R     002 SCC   = 000277
BIT04 = 000020          CRBINT= 000024          CUT1    000004R     002 NOCTSP  000302R     003 SEC   = 000261
BIT05 = 000040          CRBINX  000050R     002 CU.FST  000000RG    002 NODSRP  000314R     003 SEN   = 000270
BIT06 = 000100          CRBLDR= 000020          CU.INI  000352RG    003 NODTRC  000272R     003 SEND  = 000020
BIT07 = 000200          CRBLMX= 000652R     002 CU.IN1  000364R     003 NODTRP  000326R     003 SEV   = 000262
BIT08 = 000400          CRBLPX  000064R     002 CU.KLL  000636RG    003 NOHDC   001274R     002 SEZ   = 000264
BIT09 = 001000          CRBLX   000100R     002 CU.LST  000050RG    002 NOINTC  001262R     002 SIBLDR= 000026
BIT10 = 002000          CRBMZX= 177400          CU.MDM  000222RG    003 NOP   = 000240          SRD   = 002000
BIT11 = 004000          CRBNOX  000244R     002 CU.MOD  001206RG    002 NORCVC  001344R     002 SSYN  = 000020
BIT12 = 010000          CRBOVX  000144R     002 CU.OSU  000000RG    003 NORCVO  000720R     002 STD   = 000010
BIT13 = 020000          CRBRSX  000132R     002 CU.RST  000576RG    003 NOSRDP  000332R     003 STPSYN= 000042
BIT14 = 040000          CRBSVX  000232R     002 C.SPS = ****** G        NOSSC   001330R     002 STRSYN= 000400
BIT15 = 100000          CRBSZX= 000400          C.SREG= ****** GX       NOSTDC  000272R     003 STSSIB= 040400
CBAX  =%000005          CRBUNL= 000022          C.SRES= ****** GX       NOSYNC  001274R     002 SYNCHR= 000040
CBBX  =%000004          CRBUX   000160R     002 DEVPRS  000372R     003 NSRDPX  000326R     003 SYNIDL= 000037
CCC   = 000257          CRBXXX  000252R     002 DEVSTS= 000040          NTWP  = 000030          SYNREG= 000377
CD    = 010000          CRVOX   000140R     002 DE.PRI= ****** G        OFF   = 000000          TASKAD= ****** GX
CFINX   000250R     002 CRYOX   000214R     002 DNA   = 100000          OS    = 000000          TBUSY = 000016
CFKLU = ****** GX       CSINX   000156R     002 DNAINT= 000040          OUTCHR  001116R     002 TCOUNT= 000002
CHL68 = 002000          CTBA  = 000032          DSETC = 100000          OUTMRK  001124R     002 TDONE = 000200
CHL78 = 004000          CTBC  = 000034          DSINTE= 000040          OUTSYN  001134R     002 TFINI   001054R     002
CKRCVI  001250R     002 CTS   = 020000          DSR   = 001000          PS    = 177776          TINTE = 000100
CKSSDC  001314R     002 CUFSTP  000652RG    002 DTR   = 000002          R     = 000004          TPRIOR= 000014
CKSTSI  001262R     002 CUGRCH  000700R     002 DUCSR = ****** G        RACT  = 004000          TRANS   001010R     002
CLALL = 000257          CULSTP  000654RG    002 DUPSTS= 000046          RBUSY = 000016          TRANSI  001002R     002
CLC   = 000241          CUOSUX  000212R     003 DUSTS = 000044          RCS   = 000001          TSTSNS  001400R     002
CLN   = 000250          CUOSU1  000004R     003 ENBPAR= 001000          RCSR  = 000010          T.EFLG= ****** GX
CLRDTR  000336R     003 CUOSU2  000006R     003 ERROR   000364R     003 RCVSIB= 000000          USEPAR= 000000
CLRRI   001356R     002 CUOSU3  000044R     003 ERROR1  000360R     003 RDONE = 000200          VECTOR= 000001
CLRSS   001364R     002 CUOSU4  000116R     003 EVNPAR= 000400          RETNX =%000007          WINDUP  001152R     002
CLRSSD  001372R     002 CUOSU5  000200R     003 EXITT   001144R     002 RGOT    000666R     002 XMTSIB= 001000
CLRSTD  000344R     003 CUOSU6  000170R     003 GOTLIN  000372R     003 RING  = 040000          $RQSCH= ****** GX
CLV   = 000242          CURIS   000656RG    002 HALT  = 000000          RINTE = 000100          $SIGFL= ****** GX
CLZ   = 000244          CURXIT  000726R     002 HDUPL = 000010          RPRIOR= 000014          $TSKHD= ****** GX
CPVP  = ****** G        CUR1    000000R     002 ISRSIZ= 000050          RSX   = 000001          ..CINT= ****** GX
CP.INI  000352RG    003 CUS1    000010R     002 LINENO= 000036          RSX11M= 000001          ..DINT= ****** GX
CP.KLL  000636RG    003 CUTEXT  001170R     002 LNO   = 000000          RTSA  = 000004

. ABS.  000000      000
        000000      001
ISR     001404      002
XXX     000644      003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  1089 WORDS  ( 5 PAGES)
DYNAMIC MEMORY:  2810 WORDS  ( 10 PAGES)
ELAPSED TIME:  00:00:17
 256,020 DU11;27, 256,030 DU11/CRF/-SP=ISRRXM,OPDEF,DU11.TBL,DU11.ISR
```

DU11 CREATED BY MACRO ON 6-DEC-77 AT 08:20 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE	REFERENCES							
BIT00	= 000001	#10-26	14-433	14-439	15-484	15-487	16-583	16-584	
BIT01	= 000002	#10-27	10-51	14-441					
BIT02	= 000004	#10-28	10-52	14-451					
BIT03	= 000010	#10-29	10-53	10-68	14-454	15-489			
BIT04	= 000020	#10-30	10-54	10-69	14-433				
BIT05	= 000040	#10-31	10-55	10-70	15-484				
BIT06	= 000100	#10-32	10-56	10-71	14-433				
BIT07	= 000200	#10-33	10-57	10-72					
BIT08	= 000400	#10-34	10-58	10-73	10-79				
BIT09	= 001000	#10-35	10-59	10-80	15-484				
BIT10	= 002000	#10-36	10-60	10-81					
BIT11	= 004000	#10-37	10-61	10-74	10-82	15-484			
BIT12	= 010000	#10-38	10-62	10-74	15-484	15-498			
BIT13	= 020000	#10-39	10-63	11-173	15-500				
BIT14	= 040000	#10-40	10-64	11-171	15-503				
BIT15	= 100000	#10-41	10-65	10-75	14-433				
CBAX	=%000005	#6-216	*6-224	6-225	6-226	6-227	*6-228	*7-239	7-243	7-245
		7-252	*8-270	8-271	8-273				
CBBX	=%000004	#6-217	*7-238	7-245	*8-271	8-277	*8-279	*8-283	
CCC	= 000257	#2-10							
CD	= 010000	#10-62							
CFINX	000250 R	*6-226	8-270	*8-274	*8-276	#8-296			
CFKLU	= ****** GX	5-169							
CHL68	= 002000	#10-81	15-494						
CHL78	= 004000	#10-82	15-494						
CKRCVI	001250 R	#15-489							
CKSSDC	001314 R	#15-503	15-520						
CKSTSI	001262 R	#15-492	15-516						
CLALL	= 000257	8-280	#10-12	13-418	16-548	16-602	17-643	18-654	
CLC	= 000241	#2-6	10-12						
CLN	= 000250	#2-9	10-12						
CLRDTR	000336 R	14-442	#14-463						
CLRRI	001356 R	15-490	#15-515						
CLRSS	001364 R	15-501	#15-519						
CLRSSD	001372 R	15-504	#15-523						
CLRSTD	000344 R	#14-467							
CLV	= 000242	#2-7	10-12						
CLZ	= 000244	#2-8	10-12						
CPVP	= ****** G	4-122	5-162						
CP.INI	000352 RG	16-536	#16-537						
CP.KLL	000636 RG	18-650	#18-651						
CP.MDM	000222 RG	14-426	#14-427						
CP.MOD	001206 RG	15-476	#15-477						
CP.OSU	000000 RG	13-342	#13-343						
CP.RST	000576 RG	17-627	#17-628						
CRBINT	= 000024	#1-5	#10-108	10-109	16-589				
CRBINX	000050 R	5-168	#6-224						
CRBLDR	= 000020	#10-106	10-107	11-174					
CRBLMX	= 000652 R	*6-227	7-243	*7-245	*7-252	8-271	*8-273	#8-301	
CRBLPX	000064 R	#6-227	6-229						
CRBLX	000100 R	5-166	#7-238						
CRBMZX	= 177400	#6-211	6-224	7-248	8-276				

SYMBOL	VALUE		REFERENCES							
CRBNOX	000244	R	8-272	#8-288						
CRBOVX	000144	R	7-244	#7-252						
CRBRSX	000132	R	#7-248							
CRBSVX	000232	R	8-278	#8-283						
CRBSZX	= 000400		#6-210	6-211	7-260	8-296	8-300			
CRBUNL	= 000022		#10-107	10-108	16-590					
CRBUX	000160	R	5-167	#8-270						
CRBXXX	000252	R	#8-300							
CRVOX	000140	R	7-247	#7-249						
CRYOX	000214	R	8-275	#8-277						
CSINX	000156	R	*6-225	7-239	*7-246	*7-248	#7-260			
CTBA	= 000032		#10-111	10-112	12-233	*12-234	*12-243	*13-370	13-387	*16-592
CTBC	= 000034		#10-112	10-113	12-229	*12-235	*12-248	*13-374	*16-593	
CTS	= 020000		#10-63	12-224						
CUFSTP	000652	RG	10-48	#10-123	16-541	17-633				
CUGRCH	000700	R	#11-169							
CULSTP	000654	RG	10-48	#10-124						
CUOSUX	000212	R	13-388	13-396	13-402	13-406	#13-413			
CUOSU1	000004	R	#13-349							
CUOSU2	000006	R	#13-352							
CUOSU3	000044	R	13-361	13-364	#13-368					
CUOSU4	000116	R	13-369	#13-387						
CUOSU5	000200	R	13-390	#13-409						
CUOSU6	000170	R	13-398	#13-405						
CURIS	000656	RG	5-149	10-47	#11-149					
CURXIT	000726	R	11-165	#11-179						
CUR1	000000	R	#5-142	10-96	10-97	16-572				
CUS1	000010	R	#5-160	10-97						
CUTEXT	001170	R	12-218	12-236	12-253	12-280	12-288	12-292	12-296	#12-307
CUTIS	000736	RG	5-156	10-47	#12-207					
CUTRDN	000670	R	#11-164							
CUTSTX	000144	R	13-382	13-384	#13-395					
CUTXT1	001052	R	#12-250							
CUT1	000004	R	#5-150	10-96	16-576					
CU.FST	000000	RG	3-47	#5-141	10-123					
CU.INI	000352	RG	10-47	#16-540						
CU.IN1	000364	R	#16-547							
CU.KLL	000636	RG	10-47	#18-654						
CU.LST	000050	RG	3-47	#5-179	10-124					
CU.MDM	000222	RG	10-47	#14-430						
CU.MOD	001206	RG	10-47	#15-479						
CU.OSU	000000	RG	10-47	#13-346						
CU.RST	000576	RG	10-47	#17-631						
C.SPS	= ******	G	5-165	10-44						
C.SREG	= ******	GX	16-565							
C.SRES	= ******	GX	16-546	16-579						
DEVPRS	000372	R	#16-560							
DEVSTS	= 000040		#10-116							
DE.PRI	= ******	G	4-122	5-163						
DNA	= 100000		#10-75							
DNAINT	= 000040		#10-70	12-295						
DSETC	= 100000		#10-65							

SYMBOL	VALUE		REFERENCE							
DSINTE	= 000040		#10-55							
DSR	= 001000		#10-59	14-449						
DTR	= 000002		#10-51	14-443	14-452	14-463				
DUCSR	= ******	G	3-77	5-161						
DUPSTS	= 000046		#10-119	10-120	16-586					
DUSTS	= 000044		#10-118	10-119	16-582					
ENBPAR	= 001000		#10-80							
ERROR	000364	R	#16-548	16-563						
ERROR1	000360	R	#16-546	16-574	16-578					
EVNPAR	= 000400		#10-79							
EXITT	001144	R	12-230	12-278	#12-295					
GOTLIN	000372	R	16-543	#16-553						
HALT	= 000000		#2-4							
HDUPL	= 000010		#10-68	15-496						
ISRSIZ	= 000050		#10-120							
LINENO	= 000036		#10-113	10-114						
LNO	= 000000		#3-80	5-171						
LSI11	= ******		13-353	13-413						
MMODE	= 014000		#10-74	13-381						
MODXIT	001400	R	#15-526							
MRESET	= 000400		#10-73	16-587	17-641					
NOCLC	001266	R	#15-495							
NOCTSP	000302	R	#14-449							
NODSRP	000314	R	14-450	#14-452						
NODTRC	000272	R	14-440	#14-444	14-464					
NODTRP	000326	R	14-453	#14-455						
NOHDC	001274	R	#15-497							
NOINTC	001262	R	15-488	#15-493						
NOP	= 000240		#2-5							
NORCVC	001344	R	15-507	#15-510						
NORCVO	000720	R	11-172	#11-174						
NOSRDP	000332	R	14-434	14-446	#14-459					
NOSSC	001330	R	15-499	#15-506	15-524					
NOSTDC	000272	R	#14-445	14-468						
NOSYNC	001274	R	#15-498							
NSRDPX	000326	R	#14-456							
NTWP	= 000030		#10-110	10-111	12-242	*12-249	*13-378	*16-591		
OFF	= 000000		#4-118	5-166						
OS	= 000000		#3-70	3-72	3-75	4-124				
OUTCHR	001116	R	12-225	#12-283						
OUTMRK	001124	R	#12-287	12-300						
OUTSYN	001134	R	12-284	#12-291						
PS	= 177776		#10-15	13-357	13-358	13-416				
R	= 000004		#10-96	11-158	11-166	11-174				
RACT	= 004000		#10-61							
RBUSY	= 000016		#10-104	10-105						
RCS	= 000001		3-62	10-127	13-341	14-425	16-535	17-626	18-649	
RCSR	= 000010		#10-99	10-100	11-158	12-213	13-352	14-435	15-482	16-561	17-640
RCVSIB	= 000000		#1-4							
RDONE	= 000200		#10-57							
RETNX	=%000007		#6-218	*6-230	*7-250	*7-254	*8-281	*8-286	*8-289	
RGOT	000666	R	#11-161	11-175						

SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES								
RING	= 040000		#10-64								
RINTE	= 000100		#10-56	15-491	15-515						
RPRIOR	= 000014		#10-101	10-102	16-570						
RSTS	= ******		3-57	3-66	3-76	5-135	5-181	6-209	7-256	8-291	10-92
			11-150	11-180	11-189	12-208	12-308	12-317	16-553	17-632	
RSX	= 000001		#1-2	12-257	15-472	15-528					
RSX11D	= ******		3-69	4-110	4-117	4-121	7-240	11-192	12-245	12-259	12-314
			13-338	13-371	13-375	16-564	17-634				
RSX11M	= 000001		#1-3	3-69	4-114	4-121	11-153	11-181	11-195	12-239	12-250
			12-272	12-320	13-338	16-564	17-634				
RTSA	= 000004		#10-52	12-303	13-379	13-394	13-409				
RTSCTR	= 000015		#10-103	10-104	12-219	*12-226	*12-279	*12-299	*13-383	*16-595	
RT11	= ******		3-50	3-69	4-121	5-143	5-151	11-153	11-181	11-195	12-320
			16-564	16-597	16-605	17-634					
RVECT	= 000012		#10-100	10-101	16-569	17-635					
S	= 000010		#10-97	10-99	12-213	12-219	*12-226	12-229	12-233	*12-234	*12-235
			12-242	*12-243	*12-248	*12-249	12-277	*12-279	12-283	12-291	*12-299
SCC	= 000277		#2-15								
SEC	= 000261		#2-11								
SEN	= 000270		#2-14								
SEND	= 000020		#10-69	12-304	13-380	13-393	13-410				
SEV	= 000262		#2-12								
SEZ	= 000264		#2-13								
SIBLDR	= 000026		#10-109	10-110							
SRD	= 002000		#10-60								
SSYN	= 000020		#10-54	15-491	15-508	15-509	15-515				
STD	= 000010		#10-53	14-467							
STPSYN	= 000042		#10-117	10-118	11-166	*15-505	*15-523	*16-596			
STRSYN	= 000400		#10-58	15-502	15-505	15-519					
STSSIB	= 040400		#10-89								
SYNCHR	= 000040		#10-115	10-116	10-117	12-291	13-405	15-483	*15-511	*16-582	*16-586
			16-588								
SYNIDL	= 000037		#10-114	10-115	12-277	12-283	*13-362	*13-365	13-389	13-397	*16-594
SYNREG	= 000377		#10-78								
TASKAD	= ******	GX	12-258								
TBUSY	= 000016		#10-105	10-106							
TCOUNT	= 000002		#10-85	12-279							
TDONE	= 000200		#10-72								
TFINI	001054	R	12-244	#12-256							
TINTE	= 000100		#10-71	12-295	13-380						
TPRIOR	= 000014		#10-102	10-103	13-358						
TRANS	001010	R	#12-233								
TRANS!	001002	R	12-220	#12-229							
TSTSNS	001400	R	15-485	15-512	#15-526						
T.EFLG	= ******	GX	*12-263								
USEPAR	= 000000		10-137								
VECTOR	= 000001		#3-79	3-83	4-87	10-132					
WINDUP	001152	R	12-221	#12-299							
XMTSIB	= 001000		#10-88								
$RQSCH	= ******	GX	*12-264								
$SIGFL	= ******	GX	*12-265								
$TSKHD	= ******	GX	12-264								

SYMBOL	VALUE		REFERENCES	
. CINT	= ******	GX	16-573	16-577
. . DINT	= ******	GX	17-636	17-638

ERRHAN MACRO M1104 06-DEC-77 08:21
TABLE OF CONTENTS

2- 44 ERRHAN
3- 70 UBFHAN - USER BUFFER OVERFLOW MESSAGE
4- 95 MESSAGE AREA

```
ERRHAN  MACRO M1104  06-DEC-77 08:21  PAGE 1


                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM

                                ;
                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                        .TITLE  ERRHAN
                        .IDENT  /00002/
                ;
                ; COPYRIGHT (C)  1975, 1976, 1977
                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                ;
                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                ;
                ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                ;
                ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                ; DEC.
                ;
                ; VERSION 0
                ;
                ; ROBERT H.  ROSENBAUM 31-MARCH-75
                ;
                ; LAST MODIFICATION ON 29-MAY-75
                ;
                ; VERSION 1
                ;
                ; NOV 1977 - CEP
                ; ADD TIMEOUT COUNTERS TO ERROR PRINTOUT
                ;
                ; RECEIVE AND TRANSMIT ERROR SUMMARY PRINT OUT
                ;
000000                          .PSECT  MAIN
                ;
                ; MACRO CALLS
                                .MCALL  DIR$,QIO$
000000          QIOLBL: QIO$            ;JUST DEFINE QIO OFFSETS
        000000'                 .=QIOLBL
                ;
                                .SBTTL  ERRHAN
                ;+
                ; INPUTS:
                ;       NONE
                ; OUTPUTS:
                ;       ROUTINE WILL PRINT ON THE OPERATORS CONSOLE THE CURRENT
                ;       RECEIVE AND TRANSMIT ERROR COUNTS
                ;-
000000  012767  000000' 000014G ERRHAN:: MOV     #ERRMSG,QDPBM+Q.IOPL    ;SET BUFFER ADDRESS IN QIO DPB
000006  012702  000007'                 MOV     #RCVNUM,R2              ;PUT ADDR OF WHERE TO STORE
                                                                        ;ASCII RECV ERROR # IN R2 FOR R10OUT
000012  016700  000000G                 MOV     RCVERR,R0       ;# OF RECEIVE ERRORS
000016  004767  000000G                 JSR     PC,R10OUT       ;CONVERT IT TO ASCII
000022  012702  000026'                 MOV     #XMTNUM,R2      ;WHERE TO PUT XMT ERROR #
```

```
ERRHAN  MACRO M1104  06-DEC-77 08:21  PAGE 2-1
ERRHAN

     58 000026  016700  000000G                 MOV     XMTERR,R0          ;GET NUMBER OF XMIT ERROR
     59 000032  004767  000000G                 JSR     PC,R10OUT          ;CONVERT TO ASCII
     60 000036  012702  000045'                 MOV     #RCVNM1,R2         ;MOVE ADD TO STORE RCV TIMEOUT COUNT IN R2
     61 000042  016700  000000G                 MOV     RCVTMT,R0          ;# OF RECEIVE TIMEOUTS
     62 000046  004767  000000G                 JSR     PC,R10OUT          ;CONVERT
     63 000052  012702  000064'                 MOV     #XMTNM1,R2         ;SAME FOR TRANSMIT TIMEOUTS
     64 000056  016700  000000G                 MOV     XMTTMT,R0          ;
     65 000062  004767  000000G                 JSR     PC,R10OUT          ;
     66 000066  012767  000067  000016G         MOV     #ERRLNG,QDPBM+Q.IOPL+2  ;SET CHARACTER COUNT IN Q10 DPB
     67 000074                                  DIRS    #QDPBM             ;QUEUE THE MSG
     68 000102  000167  000000G                 JMP     CROUTX             ;PRINT A PROMPT WHEN FINISHED
```

```
ERRHAN  MACRO M1104  06-DEC-77 08:21  PAGE 3
UBFHAN - USER BUFFER OVERFLOW MESSAGE


                                                .SBTTL  UBFHAN - USER BUFFER OVERFLOW MESSAGE
                                        ;+
                                        ; CALLING SEQUENCE
                                        ;       JSR     PC,UBFHAN
                                        ;
                                        ;       CALLED AFTER A RECEIVE COMPLETE TO INFORM THE OPERATOR
                                        ;       OF ANY RECORDS THAT WERE TRUNCATED DUE TO THERE BEING
                                        ;       GREATER THAN 134 CHARACTERS IN THE RECEIVED DATA RECORD
                                        ;
                                        ; INPUTS:
                                        ;       NONE
                                        ; OUTPUTS:
                                        ;       ROUTINE PRINTS THE DECIMAL EQUIVALENT OF THE NUMBER IN
                                        ;       "UBFCNT" ON THE OPERATORS CONSOLE AND THEN ZERO'S THE
                                        ;       COUNT
                                        ;-
000106  012702  000067'          UBFHAN: MOV     #UBFMSG,R2      ;GET THE ADDRESS OF THE MESSAGE
000112  010267  000014G                  MOV     R2,QDPBM+Q.IOPL ;SET MESSAGE ADDR FOR Q10
000116  016700  000000G                  MOV     UBFCNT,R0       ;GET NUMBER OF TRUNCATED RECORDS
000122  005067  000000G                  CLR     UBFCNT          ;RESET IT
000126  004767  000000G                  JSR     PC,R10OUT       ;CONVERT IT TO RAD10
000132  012767  000025  000016G         MOV     #UBFLNG,QDPBM+Q.IOPL+2  ;SET UP CHARACTER COUNT
000140                                  DIRS    #QDPBM          ;QUEUE THE MESSAGE
000146  000207                          RTS     PC              ;RETURN
```

```
                                        .SBTTL  MESSAGE AREA
                                ;
                                        .PSECT  MESAG2,GBL
000000  122     040     116     ERRMSG: .ASCII  "R NAK: "
000003  101     113     072
000006  011
000007  040     040     040     RCVNUM: .ASCII  "        T NAK: "
000012  040     040     040
000015  040     040     124
000020  040     116     101
000023  113     072     040
000026  040     040     040     XMTNUM: .ASCII  "        R TMT: "
000031  040     040     040
000034  040     040     122
000037  040     124     115
000042  124     072     040
000045  040     040     040     RCVNM1: .ASCII  "        T TMT: "
000050  040     040     040
000053  040     040     124
000056  040     124     115
000061  124     072     040
000064  040     040     040     XMTNM1: .ASCII  "    "
        000067                  ERRLNG=.-ERRMSG
                                ;
000067  040     040     040     UBFMSG: .ASCII  "    RECORDS TRUNCATED"
000072  040     122     105
000075  103     117     122
000100  104     123     040
000103  124     122     125
000106  116     103     101
000111  124     105     104
        000025                  UBFLNG=.-UBFMSG
        000001                  .END
```

```
ERRHAN  MACRO M1104  06-DEC-77 08:21  PAGE 4-1
SYMBOL TABLE

CROUTX= ****** GX        Q10LBL  000000R     002 Q.10SB= 000010          R100UT= ****** GX        XMTERR= ****** GX
ERRHAN  000000RG    002  Q.10AE= 000012          RCVERR= ****** GX          TAGNUM= 000000          XMTNM1  000064R     003
ERRLNG= 000067           Q.10EF= 000006          RCVNM1  000045R     003 UBFCNT= ****** GX        XMTNUM  000026R     003
ERRMSG  000000R     003  Q.10FN= 000002          RCVNUM  000007R     003 UBFHAN  000106RG    002 XMTTMT= ****** GX
NRCVSF= 000000           Q.10LU= 000004          RCVTMT= ****** GX          UBFLNG= 000025          $$$ARG= 000000
NTRNSF= 000002           Q.10PL= 000014          RSX    = 000001            UBFMSG  000067R     003 $$$OST= 000014
QDPBM = ****** GX        Q.10PR= 000007          RSX11M= 000001

. ABS.   000000      000
         000000      001
MAIN     000150      002
MESAG2   000114      003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  1200 WORDS  ( 5 PAGES)
DYNAMIC MEMORY:  2810 WORDS  ( 10 PAGES)
ELAPSED TIME:  00:00:06
 256,020 ERRHAN;27, 256,030 ERRHAN/CRF/-SP=UAPRXM,ERRHAN
```

ERRHAN CREATED BY MACRO ON 6-DEC-77 AT 08:21 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES					
CROUTX	= ******	GX	2-68					
ERRHAN	000000	RG	#2-52					
ERRLNG	= 000067		2-66	#4-103				
ERRMSG	000000	R	2-52	#4-98	4-103			
NRCVSF	= 000000		#1-5					
NTRNSF	= 000002		#1-4					
QDPBM	= ******	GX	*2-52	*2-66	2-67	*3-87	*3-91	3-92
QIOLBL	000000	R	#2-41	2-42				
Q.IOFN	= 000002		2-41					
Q.IOPL	= 000014		*2-52	*2-66	*3-87	*3-91		
RCVERR	= ******	GX	2-55					
RCVNM1	000045	R	2-60	#4-101				
RCVNUM	000007	R	2-53	#4-99				
RCVTMT	= ******	GX	2-61					
RSX	= 000001		#1-2					
RSX11M	= 000001		#1-3					
R1OOUT	= ******	GX	2-56	2-59	2-62	2-65	3-90	
TAGNUM	= 000000		#1-8					
UBFCNT	= ******	GX	3-88	*3-89				
UBFHAN	000106	RG	#3-86					
UBFLNG	= 000025		3-91	#4-106				
UBFMSG	000067	R	3-86	#4-105	4-106			
XMTERR	= ******	GX	2-58					
XMTNM1	000064	R	2-63	#4-102				
XMTNUM	000026	R	2-57	#4-100				
XMTTMT	= ******	GX	2-64					
$$$ARG	= 000000		#2-41	2-41	2-41			
$$$GLB	= ******		2-41					

ERRHAN CREATED BY MACRO ON 6-DEC-77 AT 08:21 PAGE 2
MACRO CROSS REFERENCE CREF

MACRO NAME	REFERENCES		
DIR$	#2-40	2-67	3-92
ERROR	#1-9		
ERRORK	#1-16		
ERRORM	#1-32		
ERRORP	#1-47		
FATAL	#1-53		
OFF$	#2-41		
QDPB$	#2-41	2-41	
Q10$	#2-40	#2-41	

ENTER MACRO M1104 06-DEC-77 08:21 PAGE 1

```
                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM

                                ;
                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                                .TITLE  ENTER
                                .IDENT  /00004/
                        ;
                        ; COPYRIGHT (C)  1975
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                        ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                        ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                        ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                        ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                        ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                        ;
                        ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ; DEC.
                        ;
                        ; VERSION 0
                        ;
                        ; ROBERT H.  ROSENBAUM
                        ;
                        ; LAST MODIFICATION ON 31-MARCH-75
                        ;
                        ; OPEN FILES FOR WRITE ROUTINE
                        ;
000000                          .PSECT MAIN
                        ;
                        ; MACRO CALLS
                                .MCALL  OFNB$W,FDOFF$
000000                          FDOFF$  DEF$L
                        ;
                        ;+
                        ; INPUTS:
                        ;       FILE NAME BLOCK FILLED IN TO FILE NAME TO OPEN
                        ; OUTPUTS:
                        ;       ROUTINE WILL OPEN A FILE FOR WRITING AND, IF SELECTIVE
                        ;       RECEIVE IS ENABLED, IT WILL OPEN A SECOND FILE FOR WRITING
                        ;-
                                  IF DF  RSX
000000  012700  000000G        ENTER:: MOV     #FDBOU0,R0      ;GET FIRST OUTPUT FDB ADDR
000004  012767  000000G 000000G        MOV     #CHANL,SWRSUP   ;SELECT FIRST FILE FOR OUTPUT
000012  132767  000020  000000G        BITB    #20,SWRBN       ;A TASK IMAGE FILE?
000020  001407                        BEQ     5$              ;NO
000022  012760  001000  000002        MOV     #512.,F.RSIZ(R0)        ;CHANGE RECORD SIZE
000030  112760  000000G 000000        MOVB    #R.FIX,F.RTYP(R0)       ;FIXED LENGTH RECORDS
000036  000411                        BR      7$              ;OPEN THE FILE
000040  016760  000000G 000002  5$:   MOV     RVAL,F.RSIZ(R0) ;SET RECORD SIZE
000046  116760  000000G 000000        MOVB    RTYP,F.RTYP(R0) ;SET RECORD TYPE
000054  116760  000000G 000001        MOVB    RATT,F.RATT(R0) ;SET RECORD ATTRIBUTES
000062                          7$:   OFNB$W                  ;OPEN MAIN FILE FOR WRITE
000074  103427                        BCS     10$             ;RETURN IF ANY ERRORS
```

```
ENTER   MACRO M1104  06-DEC-77 08:21  PAGE 2-1


     58 000076  10576?  000000G                 TSTB    SWRSL           ;SELECTIVE RECEIVE ACTIVE?
     59 000102  00142$                           BEQ     10$             ;NO
     60 000104  012767  000000G 000000G         MOV     #CHANL2,SWRSUP  ;YES, SELECT THE "PRINTER", WHICH IS
     61                                                                 ; THE SECOND OUTPUT FILE, AS THE DEFAULT
     62                                                                 ; OUTPUT CHANNEL.
     63 000112  012700  000000G                 MOV     #FDBOU1,R0      ;GET SECONDARY FILE'S FDB ADDR
     64 000116  062700  000102                  ADD #F.FNB,R0           ;POINT TO FNB AREA
     65 000122  012701  000017                  MOV     #S.FNBW,R1      ;NUMBER OF WORDS TO MOVE
     66 000126  012702  000000G                 MOV     #DFNLP,R2       ;FROM A TRANSFER AREA
     67 000132  004767  000000G                 JSR     PC,XFRFNB       ;MOVE DATA
     68 000136  012700  000000G                 MOV     #FDBOU1,R0      ;RESET R0 FOR THE OPEN
     69 000142                                  OFNB$W                  ;YES, OPEN SECONDARY FILE
     70 000154  000207                  10$:    RTS     PC
     71                                         .ENDC
     72         000001                          .END
```

```
ENTER   MACRO M1104  06-DEC-77 08:21  PAGE 2-2
SYMBOL TABLE

CHANL = ****** GX        F.CHR = 000075        F.MBC1= 000055        F.VBN = 000064        RVAL  = ****** GX
CHANL2= ****** GX        F.CNTG= 000034        F.MBFG= 000056        F.VBSZ= 000060        R.FIX = ****** GX
DFNLP = ****** GX        F.DFNB= 000046        F.NRBD= 000024        NRCVSF= 000000        SWRBN = ****** GX
ENTER   000000RG    002  F.DSPT= 000044        F.NREC= 000030        NTRNSF= 000002        SWPSL = ****** GX
FDBOU0= ****** GX        F.DVNM= 000134        F.OVBS= 000030        N.DID = 000024        SWRSUP= ****** GX
FDBOU1= ****** GX        F.EFBK= 000010        F.RACC= 000016        N.DVNM= 000032        S.FATT= 000016
FO.WPT= ****** GX        F.EFN = 000050        F.RATT= 000001        N.FID = 000000        S.FDB = 000140
F.ACTL= 000076           F.EOBB= 000032        F.RNAM= 000034        N.FNAM= 000006        S.FNAM= 000006
F.ALOC= 000040           F.ERR = 000052        F.RCTL= 000017        N.FTYP= 000014        S.FNB = 000036
F.BBFS= 000062           F.FACC= 000043        F.RSIZ= 000002        N.FVER= 000016        S.FNBW= 000017
F.BDB = 000070           F.FFBY= 000014        F.RTYP= 000000        N.NEXT= 000022        S.FNTY= 000004
F.BGBC= 000057           F.FNAM= 000110        F.SEQN= 000100        N.STAT= 000020        S.FTYP= 000002
F.BKDN= 000026           F.FNB = 000102        F.SPDV= 000072        N.UNIT= 000034        S.NFEN= 000020
F.BKDS= 000020           F.FTYP= 000116        F.SPUN= 000074        RATT  = ****** GX     TAGNUM= 000000
F.BKEF= 000050           F.FVER= 000120        F.STBK= 000036        RSX   = 000001        XFRFNB= ****** GX
F.BKP1= 000051           F.HIBK= 000004        F.UNIT= 000136        RSX11M= 000001        .OPFNB= ****** G
F.BKST= 000024           F.LUN = 000042        F.URBD= 000020        RTYP  = ****** GX     ...TPC= 000140
F.BKVB= 000064           F.MBCT= 000054

.  ABS.   000000      000
          000000      001
MAIN     000156      002
ERRORS DETECTED:   0

VIRTUAL MEMORY USED:   1764 WORDS  ( 7 PAGES)
DYNAMIC MEMORY:   2810 WORDS  ( 10 PAGES)
ELAPSED TIME:   00:00:10
 256,020 ENTER,27, 256,030 ENTER/CRF/-SP=UAPRXM.ENTER
```

ENTER CREATED BY MACRO ON 6-DEC-77 AT 08:21 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES			
CHANL	= ******	GX	2-47			
CHANL2	= ******	GX	2-60			
DFNLP	= ******	GX	2-66			
ENTER	000000	RG	#2-46			
FDBOU0	= ******	GX	2-46			
FDBOU1	= ******	GX	2-63	2-68		
FO.WRT	= ******	GX	2-56	2-69		
F.FACC	= 000043		*2-56	*2-69		
F.FNB	= 000102		2-64			
F.RATT	= 000001		*2-55			
F.RSIZ	= 000002		*2-50	*2-53		
F.RTYP	= 000000		*2-51	*2-54		
NRCVSF	= 000000		#1-5			
NTPNSF	= 000002		#1-4			
RATT	= ******	GX	2-55			
RSX	= 000001		#1-2	2-45		
RSX11M	= 000001		#1-3			
RTYP	= ******	GX	2-54			
RVAL	= ******	GX	2-53			
R.FIX	= ******	GX	2-51			
SWRBN	= ******	GX	2-48			
SWRSL	= ******	GX	2-58			
SWRSUP	= ******	GX	*2-47	*2-60		
S.FNBW	= 000017		2-65			
TAGNUM	= 000000		#1-8			
XFRFNB	= ******	GX	2-67			
.OPFNB	= ******	G	2-56	2-56	2-69	2-69

MACRO NAME	REFERENCES									
CMOVSB	#2-56	#2-56	#2-56	#2-56	#2-56	#2-56	#2-69	#2-69	#2-69	#2-69
	#2-69	#2-69								
CMOVSW	#2-56	2-56	2-56	2-56	#2-56	2-56	2-56	#2-69	2-69	2-69
	2-69	#2-69	2-69	2-69						
ERROR	#1-9									
ERRORK	#1-16									
ERRORM	#1-32									
ERRORP	#1-47									
ERRS	#2-56	2-56	#2-69	2-69						
FATAL	#1-53									
FDOFFS	#2-35	#2-36								
FDOPSR	#2-56	#2-56	#2-69	#2-69						
FDRCSR	#2-56	2-56	#2-69	2-69						
LDFDBS	#2-56	2-56	#2-56	2-56	#2-69	2-69	#2-69	2-69		
OFNBS	#2-56	2-56	#2-69	2-69						
OFNBSW	#2-35	2-56	2-69							

FREE MACRO M1104 06-DEC-77 08:21
TABLE OF CONTENTS

2- 26 CARRIAGE CONTROL MACRO

```
FREE    MACRO M1104  06-DEC-77 08:21  PAGE 1


                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM


                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                                .TITLE  FREE
                                .IDENT  /00001/
                        ;
                        ; COPYRIGHT (C)  1975, 1976, 1977
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                        ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                        ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                        ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                        ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                        ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC
                        ;
                        ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ; DEC.
                        ;
00000!                          RSX=1
                                .SBTTL  CARRIAGE CONTROL MACRO
                        ;
                        ;       THIS IS THE CARRIAGE CONTROL MACRO
                        ;       FOR THE CARRIAGE CONTROL OF THE 2780
                        ;
                        ;
                        ;       SPECIFY LINE AND CHANNELS
                        ;
                        ;       TO INDICATE A PUNCH IN THE "SOFTWARE"
                        ;       CARRIAGE CONTROL TAPE
                                .MACRO  CCONT   $LINE,$LIST
                                Q=.
                                $E= D'$LINE*2
                                .=COLTT+$E
                                $B=0
                                .IRP    $X,<$LIST>
                                $C=0
                                $M=1
                                .REPT   1
                                $C=$C+1
                                .IIF EQ  D'$X-$C,$B=$B+$M
                                $M=$M*2
                                .ENDR
                                .ENDR
                                $B=-$B-1
                                $B=$B&376
                                .WORD   $B
                                .=Q
                                .ENDM
```

```
FREE    MACRO M1104  06-DEC-77 08:21  PAGE 3
CARRIAGE CONTROL MACRO

                                        .GLOBL  FREE
                                        .GLOBL  COLTT
                                        .IF DF  RT11
                                        .GLOBL  BUFA
                                        .GLOBL  BUFB
                                        .GLOBL  BUFC
                                        .GLOBL  BUFD
                                        .ENDC
                                        .IF DF  RT11
                                        .CSECT
                                        .ENDC
                                        .IF DF  RSX
000000                                  .PSECT  MAIN
                                        .ENDC
000000  000102                  COLTT:  .REPT   66.
                                        .WORD   376
                                        .ENDR
                                ;CARRIAGE CONTROL DEFAULT
000204                                  CCONT   8.,<2>
000204                                  CCONT   16.,<3>
000204                                  CCONT   24.,<4>
000204                                  CCONT   32.,<5>
000204                                  CCONT   40.,<6>
000204                                  CCONT   48.,<7>
000204                                  CCONT   56.,<8>
                                        .IF DF  RT11
                                BUFA:   .WORD   0
                                        .WORD   0
                                .=.+1000
                                BUFB:   .WORD   0
                                        .WORD   0
                                        .=.+1000
                                        .ENDC
000204                          FREE:
          000001                        .END
```

```
FREE    MACRO M1104  06-DEC-77 08:21  PAGE 3-1
SYMBOL TABLE

COLTT   000000RG    002 NTRNSF= 000002          RSX11M= 000001          $B    = 000176          $E    = 000160
FREE    000204RG    002 Q     = 000204R   002 TAGNUM= 000000          $C    = 000014          $M    = 010000
NRCVSF= 000000          RSX   = 000001

  ABS.  000000      000
        000000      001
MAIN    000204      002
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  631 WORDS  ( 3 PAGES)
DYNAMIC MEMORY:  2810 WORDS  ( 10 PAGES)
ELAPSED TIME:  00:00:08
 256,020 FREE;27, 256,030 FREE/CRF/-SP=UAPRXM,FREE
```

FREE CREATED BY MACRO ON 6-DEC-77 AT 08:21 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL VALUE REFERENCES

COLTT 000000 RG 3-58 #3-71 3-75 3-76 3-77 3-78 3-79 3-80 3-81
FREE 000204 RG 3-57 #3-90
NRCVSF = 000000 #1-5
NTRNSF = 000002 #1-4
Q = 000204 R #3-75 3-75 #3-76 3-76 #3-77 3-77 #3-78 3-78 #3-79
3-79 #3-80 3-80 #3-81 3-81
RSX = 000001 #1-2 #2-25 3-68
RSX11M = 000001 #1-3
RT11 = ****** 3-59 3-65 3-82
TAGNUM = 000000 #1-8
$B = 000176 #3-75 3-75 #3-75 3-75 #3-75 3-75 #3-75 3-75 #3-76
3-76 #3-76 3-76 #3-76 3-76 #3-76 3-76 #3-77 3-77
#3-77 3-77 #3-77 3-77 #3-77 3-77 #3-78 3-78 #3-78
3-78 #3-78 3-78 #3-78 3-78 #3-79 3-79 #3-79 3-79
#3-79 3-79 #3-79 3-79 #3-80 3-80 #3-80 3-80 #3-80
3-80 #3-80 3-80 #3-81 3-81 #3-81 3-81 #3-81 3-81
#3-81 3-81
$C = 000014 #3-75 3-75 #3-75 3-75 3-75 #3-75 3-75 3-75 #3-75
3-75 3-75 #3-75 3-75 3-75 #3-75 3-75 3-75 #3-75
3-75 3-75 #3-75 3-75 3-75 #3-75 3-75 3-75 #3-75
3-75 3-75 #3-75 3-75 3-75 #3-75 3-75 3-75 #3-75
3-75 #3-76 3-76 #3-76 3-76 3-76 #3-76 3-76 3-76
#3-76 3-76 3-76 #3-76 3-76 3-76 #3-76 3-76 3-76
#3-76 3-76 3-76 #3-76 3-76 3-76 #3-76 3-76 3-76
#3-76 3-76 3-76 #3-76 3-76 3-76 #3-76 3-76 3-76
#3-76 3-76 #3-77 3-77 #3-77 3-77 3-77 #3-77 3-77
3-77 #3-77 3-77 3-77 #3-77 3-77 3-77 #3-77 3-77
3-77 #3-77 3-77 3-77 #3-77 3-77 3-77 #3-77 3-77
3-77 #3-77 3-77 3-77 #3-77 3-77 3-77 #3-77 3-77
3-77 #3-77 3-77 #3-78 3-78 #3-78 3-78 3-78 #3-78
3-78 3-78 #3-78 3-78 3-78 #3-78 3-78 3-78 #3-78
3-78 3-78 #3-78 3-78 3-78 #3-78 3-78 3-78 #3-78
3-78 3-78 #3-78 3-78 #3-79 3-79 #3-79 3-79 3-79
#3-79 3-79 3-79 #3-79 3-79 3-79 #3-79 3-79 3-79
#3-79 3-79 3-79 #3-79 3-79 3-79 #3-79 3-79 3-79
#3-79 3-79 3-79 #3-79 3-79 #3-80 3-80 #3-80 3-80
3-80 #3-80 3-80 3-80 #3-80 3-80 3-80 #3-80 3-80
3-80 #3-80 3-80 3-80 #3-80 3-80 3-80 #3-80 3-80
3-80 #3-80 3-80 3-80 #3-80 3-80 #3-81 3-81 #3-81
3-81 3-81 #3-81 3-81 3-81 #3-81 3-81 3-81 #3-81
3-81 3-81 #3-81 3-81 3-81 #3-81 3-81 3-81 #3-81
3-81 3-81 #3-81 3-81 3-81 #3-81 3-81
$E = 000160 #3-75 3-75 #3-76 3-76 #3-77 3-77 #3-78 3-78 #3-79
3-79 #3-80 3-80 #3-81 3-81
$M = 010000 #3-75 3-75 #3-75 3-75 3-75 #3-75 3-75 #3-75 3-75
#3-75 3-75 #3-75 3-75 #3-75 3-75 #3-75 3-75 #3-75
3-75 #3-75 3-75 #3-75 3-75 #3-75 3-75 #3-75 #3-76
3-76 #3-76 3-76 #3-76 3-76 3-76 #3-76 3-76 #3-76

FREE CREATED BY MACRO ON 6-DEC-77 AT 08:21 PAGE 2
SYMBOL CROSS REFERENCE CREF
SYMBOL VALUE REFERENCES

3-76	#3-76	3-76	#3-76	3-76	#3-76	3-76	#3-76	3-76
#3-76	3-76	#3-76	3-76	#3-76	3-76	#3-76	#3-77	3-77
#3-77	3-77	#3-77	3-77	#3-77	3-77	3-77	#3-77	3-77
#3-77	3-77	#3-77	3-77	#3-77	3-77	#3-77	3-77	#3-77
3-77	#3-77	3-77	#3-77	3-77	#3-77	#3-78	3-78	#3-78
3-78	#3-78	3-78	#3-78	3-78	#3-78	3-78	3-78	#3-78
3-78	#3-78	3-78	#3-78	3-78	#3-78	3-78	#3-78	3-78
#3-78	3-78	#3-78	3-78	#3-78	#3-79	3-79	#3-79	3-79
#3-79	3-79	#3-79	3-79	#3-79	3-79	#3-79	3-79	3-79
#3-79	3-79	#3-79	3-79	#3-79	3-79	#3-79	3-79	#3-79
3-79	#3-79	3-79	#3-79	#3-80	3-80	#3-80	3-80	#3-80
3-80	#3-80	3-80	#3-80	3-80	#3-80	3-80	#3-80	3-80
3-80	#3-80	3-80	#3-80	3-80	#3-80	3-80	#3-80	3-80
#3-80	3-80	#3-80	#3-81	3-81	#3-81	3-81	#3-81	3-81
#3-81	3-81	#3-81	3-81	#3-81	3-81	#3-81	3-81	#3-81
3-81	3-81	#3-81	3-81	#3-81	3-81	#3-81	3-81	#3-81
3-81	#3-81							

FREE CREATED BY MACRO ON 6-DEC-77 AT 08:21 PAGE 3
MACRO CROSS REFERENCE CREF

MACRO NAME	REFERENCES							
CCONT	#2-37	3-75	3-76	3-77	3-78	3-79	3-80	3-81
ERROR	#1-9							
ERROFA	#1-16							
ERROFM	#1-32							
ERRORP	#1-47							
FATAL	#1-53							

COMTEX-MACROS MACRO M1104 06-DEC-77 08:21
TABLE OF CONTENTS

    4- 21   COMTEX MACRO DEFINITIONS
    6- 160   RSX MACRO CALLS
    8- 180   GLOBAL DEFINITIONS
   12- 197   RSX FDB AND FNB DEFINITIONS
   23- 514   SYSTEM CONTROL ROUTINE
   28- 746   COMMAND COMPLETE FROM TTY
   30- 781   TRANSMIT COMMAND
   33- 819   MASTER BID
   34- 855   RECEIVE TRANSFER COMMAND
   36- 946   TEST FOR COMMAND
   37- 977   SWITCH MODIFERS
   38-1013   TRANSPARENT SWITCH
   42-1050   TERM,DISCONNECT AND OFFLINE
   43-1121   CLEAR OUT THE TEMPORARY COMMON RECEIVE SWITCH AREA
   44-1138   TABLE FOR WRITE
   45-1189   SWITCH TABLE AND ROUTINES
   49-1273   QUERY STATUS STORAGE AREA
   50-1295   PARAMETERS
   51-1362   COMMUNICATION BUFFER STORAGE

```
                        ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                  RSX=1
000001                  RSX11M=1
000002                  NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                  NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                        ;
                        ;MACRO DEFINITIONS
000000                          TAGNUM=0
                                .MACRO  ERROR   TEXT
                                ERRORK  <TEXT>
                                JSR     PC,PROMPT
                                .ENDM
                        ;
                        ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                        ;
                                .MACRO  ERRORK  TEXT
                                TAGNUM=TAGNUM+1
                                .IRP    N,<\ TAGNUM>
                                JSR     R5,TYPMSG
                                .WORD   TAG'N
                                .WORD   TAGE'N-TAG'N
                                .PSECT  MESAG2,GBL
                        TAG'N:  .ASCII  "TEXT"
                        TAGE'N:
                                .PSECT  MAIN
                                .ENDM
                                .ENDM

                        ;
                        ; MACRO TO QUEUE A MESSAGE ONLY
                        ;
                                .MACRO  ERRORM  TEXT
                                TAGNUM=TAGNUM+1
                                .IRP    N,<\ TAGNUM>
                                JSR     R5,TYPMS1
                                .WORD   TAG'N
                                .WORD   TAGE'N-TAG'N
                                .PSECT  MESAG2,GBL
                        TAG'N:  .ASCII  "TEXT"
                        TAGE'N:
                                .PSECT  MAIN
                                .ENDM
                                .ENDM

                        ; MACRO TO QUEUE A MESSAGE AND PROMPT
                        ;
                                .MACRO  ERRORP  TEXT
                                ERRORM  <TEXT>
                                JSR     PC,PROMPT
                                .ENDM

                        ;
                                .MACRO  FATAL   TEXT
                                ERRORK  <TEXT>
                                JSR     R5,DIE
                                .ENDM
```

```
                        .TITLE  COMTEX-MACROS
                        .IDENT  /00002/
                ;
                ;COPYRIGHT 1974, 1975, 1976, 1977
                ;DIGITAL EQUIPMENT CORP.
                ;146 MAIN STREET        MAYNARD, MASS. 01754
                ;
                ;DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR
                ;RELIABILITY OF ITS SOFTWARE ON EQUIPMENT WHICH
                ;IS NOT SUPPLIED BY DIGITAL
                ;
                ; VERSION 0
                ;
                ; 23-SEPTEMBER-74
                ;
                ;VERSION 1
                ; SEP 1977 - CEP
                ; MODIFICATIONS FOR CUT CODE VERSION OF 2780
                ; LIBRARY OF COMTEX MACRO DEFINITIONS
```

```
                        .SBTTL  COMTEX MACRO DEFINITIONS
                ;
                        .MACRO  LINIT,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTINT
                        .ENDM
                ;
                        .MACRO  ASRBUF,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTPUT
                        .ENDM
                ;
                        .MACRO  PUTTC,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTPUT
                        .ENDM
                ;
                        .MACRO  PUTMC,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTPUT
                        .ENDM
                ;
                        .MACRO  GETS
                        JSR     PC EMTGET
                        .ENDM
                ;
                        .MACRO  PUTTM,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTTIM
                        .ENDM

                         MACRO  PUTD,PARLST
                MOV     #PARLST,R1
                JSR     PC,EMTPUT
                 ENDM
```

```
                        ;       COMTEX-11 278J RBT EAP
                        ;
                        ;REMOTE BATCH TERMINAL EMULATOR APPLICATION PROGRAM
                        ;RT11 AND RSX-11D/M 2780 RBT OPERATION

                        ; COPYRIGHT (C) 1975, 1976, 1977
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY
                        ; ON A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ; SOFTWARE, OR ANY COPIES THEREOF, MAY NOT BE PROVIDED OR
                        ; OTHERWISE MADE AVBAILABLE TO ANY OTHER PERSON EXCEPT
                        ; FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO THESE
                        ; LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE SOFTWARE
                        ; SHALL AT ALL TIMES REMAIN IN DEC.
                        ;
                        ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ; DEC.
                        ;


                                .IF DF  RSX
                                .IF NDF RSX11M
                                .TITLE RCSRSXD.F002
                                .ENDC
                                .ENDC
                                .IF DF  RT11
                                .TITLE  RCSRT11.F002
                                .ENDC
                                .IF DF RSX11M
                                .TITLE RCSRSXM.F001
                                .IDENT  /00130/
                                .ENDC
                        ;01/12/74
                        ;       BY LOUISE GERHART
                        ;V01-01 MODIFIED 7/10/74 TO RUN WITH RSX-11M -  RHR
                        ;V01-02 7/24/74 RHR
                        ;CHANGE EXIT ROUTINE TO DO A RESTART LINIT TO DISCONNECT FROM VECTORS
                        ;DON'T WAIT FOR MESSAGE COMPLETE TO PRINT PROMPT - JUST QUEUE MESSAGE
                        ;V01-4 THRU V01-6 RHR
                        ;DIVIDE USER PROGRAM INTO SEPARATE MODULES FOR OVERLAYING WITH RSX-11M
                        ;V01-7 10/1/74 RHR
                        ;ALLOW MULTIPLE BUFFERING UP TO 12 BUFFERS ON "GET" TO SPEED UP
                        ; SUCH DEVICES AS A CARD READER
                        ;V01-03 7/31/74 LG
                        ;ENABLE RSX11D/M TO RESUME TRANSMISSION AFTER AN RVI
                        ;FIX A BUG IN THE TERM COMMAND HANDLER
                        ; THEN PROMPT AND FINALLY READ (SIZE IS THE GOAL)!!!!
                        ;CHECK TERMINATING COMMAND CHARACTER FOR A ZERO.  IF IT IS, THEN
                        ; DON'T PRINT PROMPT AGAIN - EVENT FLAG WAS SET BECAUSE OF KILL I/O
                        ;V01-12 11/13/74 RHR
                        ;IMPLEMENT LOCAL SWITCHES ON COMMAND STRINGS.  A SWITCH WILL ONLU
```

```
RCSRSXM.F001    MACRO M1104  06-DEC-77 08:21  PAGE 5-1
COMTEX MACRO DEFINITIONS

                                        ; APPLY TO THE FILE WHICH IT FOLLOWS
                                        ;V01-13 11/20/74
                                        ;ADD "GEN" "RJE" INFO TO QUERY TYPEOUT
                                        ;V01-14 1/16/75 RHR
                                        ;FURTHER MODIFICATIONS TO LOCAL SWITCHES (SEE EDIT 12)
                                        ;V01-15 1/24/75 RHR
                                        ;REMOVE ALL HALT INSTRUCTIONS AND REPLACE WITH FATAL ERROR MESSAGES
                                        ;V01-16 2/12/75 RHR
                                        ;FIX TTD ABORT SO IT WILL WORK IN BINARY
                                        ;V01-17 2/13/75 RHR
                                        ;IMPROVE ERROR REPORTING
                                        ;V01-20 3/3/75 RHR
                                        ;IMPLEMENT ABILITY TO CHANGE DEFAULT RECEIVE FILE FROM LPO:
                                        ;V01-21 3/3/75 RHR
                                        ;INCREASE RSX STACK SPACE.  IMPLEMENT ERROR COMMAND.  INSERT CHECK FOR
                                        ;STACK POINTER OVERFLOWING
                                        ;V01-22 4/28/75 RHR
                                        ;EXPAND ERROR COMMAND TO TRANSMIT ERRORS.  IMPLEMENT BINARY SELECTIVE
                                        ;RECEIVE TO MEAN RECEIVE ALL PUNCH DATA IN BINARY BUT TRANSLATE
                                        ; ALL PRINT DATA
                                        ;V01-23 5/29/75 RHR
                                        ;IMPLEMENT RECORD TRUNCATION FEATURE AS WELL AS IMCOMPLETE RECEIVE
                                        ;V01-24 6/11/75 RHR
                                        ;ABORT TRANSMISSION IF RECEIVE AN EOT IN XMIT MODE
                                        ;V01-25 7/15/75 RHR
                                        ;FIX TRANSMISSION OF BINARY AND NONBINARY FILES COMBINED
                                        ;V01-26 7/15/75 BLR
                                        ;RT-11 V02-1 (F/B) VERSION
                                        ;LAST EDIT:      17-AUG-75       BLR
                                        ;PUT IN CHANGES TO CUT CODE
                                        ;V01-27 8/11/77 CEP
                                        ; PUT IN TEMP RECEIVE SWITCHES TO FIX PROBLEM WITH /DF
                                        ; 9/77 CEP
                                        ;V01-30
                                        ; PUT IN ERROR COUNTERS FOR TIMEOUTS
                                        ; 10/77 CEP
                                        ;


                                                .IF DF  RT11
                                                .CSECT  GET
                                                .ENDC
                                                .IF DF  RSX
                                                .IF DF  RSX11M
                                                ;
                                                ; DEFINE TASK CONTROL BLOCK OFFSETS FOR RSX11M
                                                ;
                                                .MCALL  TCBDF$
000000                                          TCBDF$
                                                .ENDC
000000                                          .PSECT
000000  000000                          DSW:    .WORD   0
000000                                          .PSECT ISR
000000                                          .PSECT  MAIN

                                                .IF NDF NTRNSF
                                                NTRNSF=5                ;NUMBER OF TRANSMIT FILES IN A COMMAND
                                                                        ;STRING -1 (DEFAULT CASE =6)
```

```
                                        .ENDC
                                        .IF NDF NRCVSF
                                        NRCVSF=2                ;NUMBER OF RECEIVE FILES IN A COMMAND
                                                                ; STRING - 1 (DEFAULT = 3)
                                        .ENDC
                                        .ENDC

                                        .IF DF RSX
                                ;DEFINITIONS OF REGISTERS, ETC.

177776                                  PS=177776               ;ADDRESS OF PROC.  REGISTER
000000                                  R0=%0
000001                                  R1=%1
000002                                  R2=%2
000003                                  R3=%3
000004                                  R4=%4
000005                                  R5=%5
000006                                  SP=%6
000007                                  PC=%7
                                        .ENDC


                                ; DECLARE GLOBAL ALL EXTERNAL REFERENCES
                                        .GLOBL  INITAL,LUKUP,CLOSER,CLOSET
                                        .GLOBL  FSTGET,FSTFIL,POINT
                                        .GLOBL  MSGDON,ASROUT,GOABRT
                                        .GLOBL  QUEHAN,INPCPL,ERRHAN
                                        .GLOBL  TER1,DONE
                                        .GLOBL  ACK,NAK
                                        .IF DF  RT11
                                        .GLOBL  MSWIT,DFNBUF,LOKBLK
                                        .ENDC

000040                          ACK=40          ;PARAMETER FOR PUTD TO BISYNC TAP
000100                          NAK=100
                                        .IF DF  RSX
000001                          UIBEFN=1                        ;EFN 1 = UIB LOADED
000002                          TTPRFN=2                        ;EFN 2 = TTY PROMPT WRITE COMPLETE
000004                          TTKLFN=4                        ;EFN 3 = TTY KILL I/O COMPLETE
000010                          TTRDFN=10                       ;EFN 4 = TTY READ COMPLETE
000020                          TTERFN=20                       ;EFN 5 = TTY ERROR MESSAGE WRITE COMPLETE
000040                          SIBEFN=40                       ;EFN 6 =SIB ENTERY
                                        .ENDC
```

```
RCSRSXM.F001    MACRO M1104  06-DEC-77 08:21  PAGE 6
COMTEX MACRO DEFINITIONS

                                        .IF DF  RSX
                                        .SBTTL  RSX MACRO CALLS
                                        .MCALL  CSI$
000000                                  CSI$
                                        .MCALL  WTLO$S,CLOSE$,WTSE$S
                                        .MCALL  FDBDF$,FDAT$A,FDRC$A,FDBK$A,FDOP$A,FDBF$A,FSRSZ$
                                        .MCALL  NMBLK$,FDAT$R
                                        .MCALL  CSI$1,CSI$2
                                        .MCALL  QIO$,DIR$
                                        .MCALL  CALL
                                        .MCALL  SVTK$
                                        .ENDC
                                        .IF DF  RT11
                                        .SBTTL  RT11 MACRO CALLS
                                        .MCALL  .DSTAT,.HRESET,.TTYOUT,.EXIT,.TTINR
                                        .MCALL  .CSISPC,.PRINT,.WAIT,.CLOSE,.V2.,.SPND
                                        .MCALL  .TLOCK,.UNLOCK
                                        .V2.
                                        .ENDC
```

GLOBAL DEFINITIONS

```
                        .SBTTL  GLOBAL  DEFINITIONS
                        .IF DF  RT11
                        .GLOBL  CDEMT
                        .GLOBL  BUFB
                        .GLOBL  BUFA
                        .ENDC
                        .GLOBL  COLTT
                        .IF DF  RSX
                        .GLOBL  .FIND
                        .GLOBL  .PARSE
                        .GLOBL  C.SPS
                        .ENDC
```

```
                                        .IF DF  RSX
                                        .SBTTL  RSX FDB AND FNB DEFINITIONS
000000                          DFBLK:: NMBLK$  RJE,RJE,,SY,0   ;DEFINE DEFAULT NAME BLOCK
000036                          FDBIN:. FDBDF$                  ;DEFINE FDB FOR INPUT
000176                                  FDRC$A  FD.PLC,EXPREC,134.
000176                                  FDBF$A  ,,1
000176                                  FDOP$A  3,,DFBLK
000176                          FDBOU0::FDBDF$                  ;DEFINE FDB FOR OUTPUT
000336                                  FDRC$A  FD.PLC,EXPREC,134.
000336                                  FDBF$A  ,,2
000336                                  FDAT$A  R.VAR,FD.CR,80.
000336                                  FDOP$A  4,,DFBLK
000336                          FDBOU1::FDBDF$
                                ;
                                ;THE SECOND FDB IS ONLY USED WHEN SELECTIVE RECEIVE IS ENABLED
                                ;IN SUCH A CASE, FDBOU0 CONTAINS THE FILE NAME FOUND IN THE
                                ;COMMAND STRING  AND ALL RECORDS RECEIVED PREFACED BY AN ESC 4
                                ;ARE WRITTEN TO FDBOU0.  ALL OTHERS ARE WRITTEN TO FDBOU1.
                                ;BY IMPLEMENTATION, THE SECOND FDB IS ALWAYS FOR THE
                                ;LINE PRINTER HENCE THE FOLLOWING CODE DEFINES THE DEFAULT NAME
                                ;BLOCK AS WELL AS FILLING IN THE FILE NAME PORTION OF THE
                                ;SECOND FDB AT ASSEMBLY TIME TO BE THE LINE PRINTER.
                                ;
000476                                  FDRC$A  FD.PLC,EXPREC,134.
000476                                  FDBF$A  ,,2
000476                                  FDAT$A  R.VAR,FD.CR,80.
000476                                  FDOP$A  5,,DFNLP
000476                          DFNLP:: NMBLK$  RJE,RJE,,LP,0   ;DEFINE THE DEFAULT NAME BLOCK FOR OUTPUT
000534                          QDPBW:  QIO$    IO.WLB,2,2,,,,<PRMTA,,'$>
000564                          QDPBR:  QIO$    IO.RLB,2,4,,IOST,,<KBIBUF,80.,>
000614                          QDPBK:  QIO$    IO.KIL,2,3
000644                          QDPBM:: QIO$    IO.WLB,2,5,,,,<,,40>
000674                                  FSRSZ$  2,,MAIN
000674                          IOST:   .BLKW   2               ;IO STATUS BLOCK
000700  000000                  POINT1::.WORD   0               ;NEXT 3 LOCATIONS USED BY MARK AND POINT
000702  000000                  POINT2::.WORD   0
000704  000000                  POINT3::.WORD   0
                                ;
000706                          SSTVEC: SVTK$   TABSST,7        ;
                                ;
001332'                         TABSST::.WORD   ODDADD
001346'                                 .WORD   MEMPRO
001362'                                 .WORD   BPTINS
001376'                                 .WORD   IOTINS
001412'                                  WORD   RSVDIN
001426'                                 .WORD   ILLINS
001442'                                 .WORD   TRPINS
                                ;
                                ;
                                        .ENDC
```

```
                                ;ASSEMBLY TIME PARAMETERS


                                        .GLOBL  LPSIZE,RCVTIM,BASELN,COMAV,CMBMAX
000206                                  LPSIZE=134.             ;COMM. RCVE BUFFER SIZE

000000                                  BASELN=0                ;LINE NUMBER FOR PARAMETER LISTS

000007                                  COMAV=7                 ;NO. OF COMM BUFFERS AVAILABLE
000016                                  CMBMAX=COMAV+COMAV      ;MAX INDEX FOR COMM BUFFER LIST
000177                                  RCVTIM=127.             ;NUMBER OF SECONDS FOR RECEIVE TIMER

                                ;ASCII DEFINITIONS
                                        .GLOBL  ESC,BINITB,BINEOT,XITB,XSTEQ
                                        .GLOBL  BINTTD,BINESC,BIN4,XEOTS

000033                                  ESC=33

026402                                  BINTTD=26402            ;EBCDIC STX-ENQ FOR BINARY MODE (/2780)
000047                                  BINESC=047              ;EBCDIC ESC
000364                                  BIN4=364                ;EBCDIC 4
000037                                  BINITB=37               ;EBCDIC ITB FOR BINARY MODE (/2780)
177467                                  BINEOT=177467           ;EBCDIC EOT PAD FOR BINARY MODE (/2780)
077405                                  XCEP=77405
077404                                  XEOTS=77404
002020                                  XDEOT=2020
000177                                  XPAD=177
000037                                  XITB=37
002402                                  XSTEQ=2402
000015                                  CR=15
000012                                  LF=12
```

```
                                        ;RBT SYSTEM INITIALIZATION
                                        ;
                                        ;LINE TABLE INITIALIZATION
                                        ;
                                                .GLOBL  STKSET,LININT,RCVERR,XMTERR,UBFCNT
                                                .GLOBL  RCVTMT,XMTTMT
                                        
                                                .IF DF  RSX
000732  125252                          STKLMT: .WORD   125252  ;STACK LIMIT WORD
        001174'                                 .=.+240
                                                .ENDC
001174                                  STKSET:                 ;STACK POINTER START DEFINITION
001174  000000                          RCVERR: .WORD   0       ;ALSO FIRST WORD CONTAINS RECEIVE BLOCK ERROR COUNT
001176  000000                          XMTERR: .WORD   0       ;TRANSMIT BLOCK ERROR COUNT
001200  000000                          RCVTMT: .WORD   0       ;RECEIVE TIMEOUTS
001202  000000                          XMTTMT: .WORD   0       ;TRANSMIT TIMEOUTS
001204  000000                          UBFCNT: .WORD   0       ;USER BUFFER OVERFLOW COUNTER.  INCREMENTED
                                                                ;WHEN TAP RECEIVES A RECORD > 134 CHARACTERS
                                        
001206                                  LININT:
001206  012706  001174'                        MOV     #STKSET,SP      ;SET UP STACK POINTER
                                                .IF DF RSX
001212                                          DIRS    #SSTVEC
001220  103001                                  BCC     5$
001222  000000                                  HALT
001224                                  5$:
                                                .ENDC
001224  004767  000000G                        JSR     PC,INITAL       ;INITIALIZE THE WORLD
001230  000167  002342                        JMP     OFHAN1          ;CLEAR SOME SWITCHES
                                        ;
                                        ;
                                        ;       EMT CLLS TO REPLACE CDEMT IN SCIP
                                        ;
                                                .GLOBL  EMTPUT,EMTGET,EMTINT,EMTTIM
                                        ;
001234  004567  000000G                 EMTPUT: JSR     R5,C.SREG               ;SAVE REGISTERS
001240  004767  000000G                         JSR     PC,C.SPUT               ;
001244  100425                                  BMI     EMTERR                  ;ERROR
001246  004567  000000G                 EMTGD:  JSR     R5,C.SRES               ;RESTORE REGISTERS
001252  000257                                  CCC                             ;CLEAR COND CODES
001254  000207                                  RTS     PC                      ;RETURN
001256  004567  000000G                 EMTGET: JSR     R5,C.SREG
001262  004767  000000G                         JSR     PC,C.SGET
001266  100414                                  BMI     EMTERR
001270  000766                                  BR      EMTGD
001272  004567  000000G                 EMTINT: JSR     R5,C.SREG
001276  004767  000000G                         JSR     PC,C.STIN
001302  100406                                  BMI     EMTERR
001304  000760                                  BR      EMTGD
001306  004567  000000G                 EMTTIM. JSR     R5,C.SREG
001312  004767  000000G                         JSR     PC,CTEMT
001316  100353                                  BPL     EMTGD
001320  004567  000000G                 EMTERR: JSR     R5,C.SRES               ;RESTORE REGISTERS
001324  000257                                  CCC                             ;CLEAR COND CODES
001326  000270                                  SEN                             ;INDICATE ERROR
001330  000207                                  RTS     PC                      ;RETURN
```

```
                                ;
                                        .IF DF RSX
001332                          ODDADD: FATAL   <ODD ADDR>
001346                          MEMPRO: FATAL   <MEM PROT>
001362                          BPTINS: FATAL   <BPT INS>
001376                          IOTINS: FATAL   <IOT INS>
001412                          RSVDIN: FATAL   <RSVD INS>
001426                          ILLINS: FATAL   <ILL INS>
001442                          TRPINS: FATAL   <TRAP INS>
                                        .ENDC
```

```
                                        ;$$$$$$$$$$$$$$$$$$$$

                                        ;CONTROL WORD FLAG DEFINITIONS



                                        .IF DF RT11
                                        .GLOBL  CHEXT,DFEXT,DEVRT,DEFALT,DEFEXT,XFRFNB

                                CHEXT:  .WORD   0
                                        .WORD   0

                                DFEXT:  .WORD   0
                                        .WORD   0
                                ;
                                DEVRT:  .RAD50/RT/              ;BISYNC DEVICE NAME

                                DEFALT: .RAD50  /LP/            ;DEFAULT RECEIVE FILE
                                        .WORD   0,0,0,0
                                DEFEXT: .RAD50  /RJE/
                                        .RAD50  /RJE/           ;DEFAULT FOR OUTPUT FILES
                                        .RAD50  /RJE/
                                        .RAD50  /RJE/
                                        .ENDC
                                        .IF DF  RSX
                                        .GLOBL  DEVRT
001456      122     124         DEVRT:  .ASCII  'RT'
                                        ENDC
```

```
                                ;COMTEX 2780 RBT EAP CONTROL/STATUS FLAGS
                                ;AND SWITCHES

                                        .GLOBL  ONLINE,CODE,FLAGS,MCB,TVAB,MODE
                                        .GLOBL  HFCACT,INITF,ACKPEN,ELATCH,ENDPEN
                                        .GLOBL  TDONE,EOTPLN,ETXFLG,EOTFLG,RVIPEN
                                        .GLOBL  CRBFAV,CMBFAV,CMBFPN,ASSGNX
                                        .GLOBL  STOREX,PRINTX,IOERR,TXPEN,INPCNT
                                        .GLOBL  NAKCNT,RECSIZ,SWRSUP,CRBASX,OPNFLG,INCRCV
                                        .GLOBL  RCVCNT
                                ;
001460  000             ONLINE: .BYTE   0               ;BIT 7 = 0 OFFLINE
                                ;                       ;      = ! ONLINE
001461  010             CODE:   .BYTE   10              ;BITS 3-0 = 00 USASCII
                                ;                       ;         = 10 EBCDIC
001462  140             FLAGS:  .BYTE   140             ;BITS 1,2 - XPAR
                                ;                       ;BITS 3,4 - AUTO
                                ;                       ;BITS 5,6 - ITB
                                ;                       ;BIT 7 - SLAVE/MASTER

                                ;STATUS

001463  000             MCB:    .BYTE   0               ;MASTER CONTROL BYTE
                                ;                       ;BIT 7(1) - GEN. RECRD FORMAT
                                ;                       ;         (0) - 2780 RECRD FORMAT

001464  000             TVAB:   .BYTE   0               ;BIT 7(1) - ABORT TRANSMISSION
001465  000             MODE:   .BYTE   0               ;BITS 1-0 = 00 CONTROL
                                ;                       ;         = 01 RECEIVE
                                ;                       ;         = 10 TRANSMIT
                                ;                       ;         = 11 NOT USED
001466  000             HFCACT: .BYTE   0               ;BIT 7 = 0 NO DYNAMIC HFC
                                ;                       ;      = 1 HFC ACTIVE
001467  000             INITF:  .BYTE   0               ;BIT 7 = 0 NOT IN INIT
                                ;                       ;      = 1 IN INIT
001470  000             ACKPEN: .BYTE   0               ;BIT 7 = 0 ACK NOT PENDING
                                ;                       ;      = 1 ACK PENDING
001471  000             ELATCH: .BYTE   0               ;BIT 7 = 1 PROHIBIT SUBSEQUENT READS
001472  000             ENDPEN: .BYTE   0               ;BIT 7 = 0 END NOT PENDING
                                ;                       ;      = 1 END PENDING
001473  000             TDONE:  .BYTE   0               ;BIT 7 = 1 X-MIT DONE
001474  000             EOTPEN: .BYTE   0               ;BIT 7 = 0 EOT NOT PENDING
                                ;                       ;      = 1 EOT PENDING
001475  000             ETXFLG: .BYTE   0               ;BIT 7 = 0 ETX NOT PRESENTED
                                ;                       ;      = 1 ETX PRESENTED
001476  000             EOTFLG: .BYTE   0               ;BIT 7 = 0 EOT NOT PRESENTED
                                ;                       ;      = 1 EOT PRESENTED
001477  000             INCRCV: .BYTE   0


001500  000             RVIPEN: .BYTE   0               ;RVI PENDING
001501  007             CRBFAV: .BYTE   7               ;BITS 1-0 = 00 CR BUFFER NOT AVAILABLE
                                ;                       ;         = 01 ONE AVAILABLE
                                ;                       ;
                                ;                       ;         = 07 SEVEN AVAILABLE
001502  007             CMBFAV: .BYTE   COMAV           ;COMM BUFFERS CURRENTLY AVAILABLE
```

```
001503      000               CMBFPN: .BYTE   0             ;NUMBER OF COMM BUFFERS PENDING

001504      377               TRRTNF: .BYTE   377           ;BITS 7-0 = 0 XMIT/RCV CMD USE OF CODE
                                                            ;         = 377 OTHER USES OF CODE


001505      000               ASSGNX: .BYTE   0             ;ASSIGN SLOT INDEX (BISYNC ASRBUF)
001506      000               STOREX: .BYTE   0             ;PRINT QUEUE STORE INDEX
001507      000               PRINTX: .BYTE   0             ;PRINT SLOT INDEX


001510      000               IOERR:  .BYTE   0             ;200 = I/O ERROR
001511      000               TXPEN:  .BYTE   0             ;200 = X-MIT PENDING

001512      000               RCVCNT: .BYTE   0             ;RECEIVE TIMEOUT COUNTER
                                                            ;IF THIS GOES TO ZERO THE RECEIVER IS HUNG
001513      000               OPNFLG: .BYTE   0             ;200 = HAVE SEEN ENQ BUT HAVE NOT
                                                            ;YET OPENED THE FILE

                                      .IF DF  RT11
                                      .GLOBL  ZERFLG
                              ZERFLG: .BYTE   0             ;200  = ZERO LENGTH FILE WAS DELETED
                                      .ENDC
                                      EVEN

001514  000000                INPCNT: .WORD   0             ;INPUT COUNTER
001516  000000                NAKCNT: .WORD   0             ;INCORRECT BCC RCVE RECORD-COUNTER
001520  000120                RECSIZ: .WORD   80            ;BUFFER SIZE 80 OR 132 DECIMAL
001522  000000                SWRSUP: .WORD   0
001524  [illegible]           CRBASX: .WORD   CMBUF         ;START OF CR BUFFER SPACE
                                      .IF DF  RT11
                                      .GLOBL  STSBLK,HICORE
                              HICORE: .WORD   STSBLK        ;ADDRESS OF HI CORE
                                      .ENDC
```

```
                                        ; COMMUNICATIONS BUFFFER LIST

                                                .GLOBL  CMBLST,CMPLST,CMCLST,CMFLST
001526  004214'                         CMBLST. .WORD   CMBUF1
001530  004422'                                 .WORD   CMBUF2
001532  004630'                                 .WORD   CMBUF3
001534  005036'                                 .WORD   CMBUF4
001536  005244'                                 .WORD   CMBUF5
001540  005452'                                 .WORD   CMBUF6
001542  005660'                                 .WORD   CMBUF7

                                        ; ONLINE PRINT QUEUE - BUFFER START ADDRESSES
001544  000000                          CMPLST: .WORD   0
001546  000000                                  .WORD   0
001550  000000                                  .WORD   0
001552  000000                                  .WORD   0
001554  000000                                  .WORD   0
001556  000000                                  .WORD   0
001560  000000                                  .WORD   0

                                        ; ONLINE PRINT QUEUE - BUFFER BYTE COUNTS
001562  000000                          CMCLST: .WORD   0
001564  000000                                  .WORD   0
001566  000000                                  .WORD   0
001570  000000                                  .WORD   0
001572  000000                                  .WORD   0
001574  000000                                  .WORD   0
001576  000000                                  .WORD   0

                                        ; ONLINE PRINT QUEUE -END OF FILE
001600  000000                          CMFLST: .WORD   0
001602  000000                                  .WORD   0
001604  000000                                  .WORD   0
001606  000000                                  .WORD   0
001610  000000                                  .WORD   0
001612  000000                                  .WORD   0
001614  000000                                  .WORD   0
```

```
                                        .SBTTL  SYSTEM CONTROL ROUTINE

                                ;GETS LOOP FOR UIB ENTRY RETRIEVAL

                                        .GLOBL  CROUT,GETENT,ENTRY,TRRTNF,C.SPS

                                        .IF DF  RT11
                                        .GLOBL  GOTGET
                                GOTGET: TSTB    TRRTNF          ;PRINT THE PROMPT
                                        BMI     GETENT          ;NO
                                        COMB    TRRTNF
                                CROUT:
                                        MOV     #CR,R0          ;PRINT A CARRAIGE RETURN
                                        .TTYOUT
                                        MOV     #LF,R0          ;PRINT A LINE FEED
                                        .TTYOUT
                                        MOV     #'*,R0
                                        TSTB    ONLINE          ;ONLINE
                                        BEQ     COUT1           ;NO - ONLY ONE * ON PROMPT
                                        .TTYOUT
                                COUT1:  .TTYOUT
                                        BR      GETENT
                                        .ENDC
                                        .IF DF  RSX
001616                          GOTGET::
001616                          CROUTX::CALL    PROMPT          ;PROMPT THE OPERATOR
001622                          CROUT::
001622                          USRTIM::
001622                          GETENT::
001622  022767  125252  177102          CMP     #125252,STKLMT  ;OVERFLOWED THE STACK?
001630  001411                          BEQ     GET01           ;NO
001632                                  ERROR   <STACK OVERFLOW>        ;IT HAS
001646  012767  125252  177056          MOV     #125252,STKLMT  ;RESET STACK LIMIT WORD
001654                          GET01:
                                        .IF NDF RSX11M
                                        MOV     .CRTSK,R1       ;GET THIS TASK'S ATL NODE
                                        BIT     #SIBEFN,A.EF(R1)        ;IS 'SIB ENTRY' SET?
                                        BEQ     5$                      ;NO
                                        BIC     #SIBEFN,A.EF(R1)        ;
                                        MOV     C.STAB,R2               ;SET UP LINE TABLE ADDR
                                        JSR     PC,C.BODE               ;GO DO TRANSMIT DONE
                                        BR      10$
                                5$:     BIC     #SIBEFN,A.EF(R1)
                                10$:
                                        MOV     .CRTSK,R1       ;MAKE SURE R1 SET RIGHT - CLOBBERED BY
                                                                ;C.BODE ROUTINE
                                        BIC     #UIBEFN,A.EF(R1)        ;CLEAR UIB LOADED EVENT FLAG
                                        .IFF
001654  016701  000000G                 MOV     TASKAD,R1       ;GET THIS TASK'S TCB ADDRESS
001660  032761  000040  000022          BIT     #SIBEFN,T.EFLG(R1)      ;IS 'SIB ENTRY' SET?
001666  001410                          BEQ     5$                      ;NO
001670  042761  000040  000022          BIC     #SIBEFN,T.EFLG(R1)      ;YES - CLEAR IT
001676  016702  000000G                 MOV     C.STAB,R2               ;SET UP LINE TABLE ADD
001702  004767  000000G                 JSR     PC,C.BODE               ;GO DO TRANSMIT DONE
001706  000403                          BR      10$                     ;
001710                          5$:
001710  042761  000040  000022          BIC     #SIBEFN,T.EFLG(R1)      ;CLEAR THE EVENT FLAG
```

```
571 001716  016701  000000G          10$:    MOV     TASKAD,R1               ;RESET R1 CLOBBERED BY C.BODE
572 001722  042761  000C01  000022          BIC     #UIBEFN,T.EFLG(R1)      ;CLEAR UIB EVENT FLAG
573                                          .ENDC
574                                          .ENDC
575                                          .IF DF  RT11
576                                  GETENT: .UNLOCK                         ;RELEASE USR
577                                          JSR     PC,C.SPS                ;CHECK FOR SIB ENTRY
578                                          .ENDC
579 001730  005067  000030                   CLR     ENTRY                   ;POINTER TO PARAM LIST IN R1
580 001734  012701  001764'                  MOV     #ENTRY,R1               ;
581 001740                                   GETS                            ;GO GET ENTRY - WAS THERE ONE?
582 001744  100413                           BMI     GETEXP                  ;NO,
583 001746  116705  000013                   MOVB    ENTRY+1,R5              ;YES, SAVE CODE BYTE IN R5
584 001752  042705  177700                   BIC     #177700,R5              ;CLEAR ALL BUT 6 LOW ORDER BITS
585 001756  006305                           ASL     R5                      ;YES, MAKE INTO WORD INDEX
586 001760  000175  003136'                  JMP     @JMPTBL(R5)             ;GO THRU PRIMARY JUMP TABLE
587
588
589 001764     000     000           ENTRY:  .BYTE   0,0                     ;1ST WORD OF ENTRY
590 001766  000000                           .WORD   0                       ;2ND WORD OF ENTRY
591 001770  000000                           .WORD   0                       ;3RD WORD OF ENTRY
592 001772  000000                           .WORD   0                       ;4TH WORD OF ENTRY
593
594
595
596
597                                          ;SYSTEM CONTROL ROUTINE EXTENSION
598                                          ;FOR  I/O
599
600 001774                           GETEXP: .IF DF  RT11
601                                          .TLOCK                          ;IS USR AVAILABLE?
602                                          BCS     WAIT                    ;NO - DON'T EVEN LOOK FOR CONSOLE INPUT
603                                          MOV     #KBIBUF,R1
604                                          MOV     R1,R2
605                                  TERMIN: .TTINR                          ;TRY FOR TTY INPUT COMMAND
606                                          BCS     WAIT                    ;NO TTY INPUT - JUST WAIT FOR AN EVENT
607                                          MOVB    R0,(R1)+
608                                          CMPB    R0 #LF
609                                          BEQ     COMCPL                  ;END OF INPUT COMMAND - HANDLE COMMAND
610                                          BR      TERMIN                  ;LOOP FOR MORE INPUT
611                                          .ENDC
612                                          .IF DF  RSX
613                                          .IF NDF RSX11M
614                                          MOV     .CRTSK,R1               ;RESTORE TASKS ATL NODE - KILLED BY GETS
615                                          MOV     A.EF(R1),R2             ;GET EVENT FLAGS 1-16
616                                          .IFF
617 001774  016701  000000G                  MOV     TASKAD,R1               ;RESTORE TASKS TCB ADD - KILLED BY GETS
618 002000  016102  000022                   MOV     T.EFLG(R1),R2           ;GET EVENT FLAGS 1-16
619                                          .ENDC
620 002004  032702  000010                   BIT     #TTRDFN,R2              ;TTY READ COMPLETE?
621 002010  001423                           BEQ     SLEEP                   ;IF EQ NO - GO TO SLEEP
622                                          .IF NDF RSX11M
623                                          BIC     #TTRDFN,A.EF(R1) ;CLEAR THE FLAG (NO RACE IF TTY LOWER PRIORITY)
624                                          .ENDC
625                                          .IF DF RSX11M
626 002012  042761  000010  000022           BIC     #TTRDFN,T.EFLG(R1)
627                                          .ENDC
```

```
                                        TSTB    IOST            ;WAS THERE AN ERROR (ABORT OR  Z)?
                                        BMI     IO$             ;IS MINUS - YES, SEE WHAT KIND OF ERROR
                                        TSTB    IOST+1          ;NO ERROR - CHECK TERMINATING CHARACTER
                                        BNE     COMCPL          ;IF IT IS ZERO THE EVENT FLAG WAS SET
                                                                ;DUE TO KILL I/O OTHERWISE SET BY INPUT
                                IO$:    CMPB    #IE.EOF,IOST    ;CONTROL Z?
                                        BNE     GETENT
                                        FATAL   < EXIT >        ;YES, EXIT


                                SLEEP:
                                        WTLO$S  0,#UIBEFN!TTRDFN!SIBEFN ;WAIT FOR ANY
                                                                ;OF EFN'S 1,4,6
                                        BCC     GETENT          ;SOMETHING HAPPENED - GO FIND OUT WHAT
                                SYSERR: FATAL   <FATAL DIRECTIVE ERROR> ;ERROR IN DIRECTIVE
                                ;
                                ;ROUTINE TO TYPE MESSAGES TO THE OPERATOR
                                ;CALL:  JSR     R5,TYPMSG
                                ;       .WORD   ADDRESS OF ASCII TEXT
                                ;       .WORD   LENGTH IN BYTES
                                ;
                                ; ALTERNATE ENTRY POINT - TYPMS1
                                ; ENTER AT THIS POINT TO SET UP TO QUEUE A MSG WITHOUT
                                ; PERFORMING A KILL I/O
                                TYPMSG::DIR$    #QDPBK          ;KILL THE PENDING READ
                                        .IF NDF RSX11M
                                        WTSE$S  #3              ;THIS IS ONLY NECESSARY FOR RSX-D SINCE
                                                                ; THE TASK IS NORMALLY RUNNING AT A HIGHER
                                                                ; PRIORITY THEN THE TELETYPE HANDLER AND
                                                                ; THE KILL WOULD ALSO KILL THE MESSAGE AND
                                                                ; THE PROMPT
                                        .ENDC
                                TYPMS1::MOV     #QDPBM,R0       ;GET THE QIO PARAMETER LIST
                                        MOV     (R5)+,Q.IOPL(R0)        ;ADDRESS OF MESSAGE
                                        MOV     (R5)+,Q.IOPL+2(R0)      ;LENGTH IN BYTES
                                        DIR$    R0              ;QUEUE THE MESSAGE
                                        RTS     R5              ;RETURN

                                ;ROUTINE TO PRINT THE PROMPT AND QUEUE A READ
                                ;CALL:  JSR     PC,PROMPT

                                PROMPT::MOV     #QDPBW,R0       ;THIS IS ADDR OF QIO DPB
                                        MOV     #PRMTL,Q.IOPL+2(R0)     ;LENGTH OF PROMPT LINE
                                        TSTB    ONLINE          ;ONLINE?
                                        BPL     10$             ;NO
                                        INC     Q.IOPL+2(R0)    ;YES, PRINT EXTRA *
                                10$:    DIR$    R0              ;PRINT THE PROMPT
                                        BCS     SYSERR          ;ERROR FROM DIRECTIVE
                                        DIR$    #QDPBR          ;DO A READ
                                        BCS     SYSERR          ;DIRECTIVE ERROR
                                        RTS     PC
                                        .ENDC

                                ;ROUTINE TO HANDLE FATAL ERROR MESSAGES
                                ;CALL:  FIRST TO "TYPMSG"
                                ;       JSR     R5,DIE
                                ;
```

```
                                        .GLOBL  DIE
002206  004767  000000G       DIE:      JSR     PC,DONE         ;FINISH UP AND EXIT
                                                                ;***NO RETURN***

                                        .IF DF  RT11
                                ;
                                ;NOTHING TO DO -- TIMED WAIT THEN RETRY EVENT/TTY LOOP
                                ;
                                WAIT:   .UNLOCK                 ;INSURE THAT THE USR IS UNLOCKED
                                        TSTB    WAITNG          ;IS THERE A PENDING TIMER?
                                        BMI     20$             ;YES - DON'T SET ANOTHER
                                        MOV     #TTYWAT,-(SP)   ;MOVE PARAMETER PTR ONTO STACK
                                        PUTTM                   ;CALL SCIP TO LINK TIMER REQUEST
                                        BPL     10$             ;OK - SCIP SUCEEDED IN QUEUEING TIMER
                                        BR      WAIT            ;NOT ENOUGH QUEUE ELEMENTS - RETRY
                                10$:    BISB    #377,WAITNG     ;INDICATE TIMED WAIT IN PROGRESS
                                20$:    .SPND                   ;SUSPEND OPERATION
                                        BR      GETENT          ;JUST LOOP TO TEST WHEN EVENT OCCURS
                                ; USER TIMER HAS EXPIRED
                                ;
                                USRTIM: .IF DF  RT11    CLRB    WAITNG  ;CLEAR TIMER PENDING FLAG
                                        TSTB    RCVCNT          ;IS THE RECEIVE TIMER SET?
                                        BEQ     GETENT          ;NO, GET NEXT UIB ENTRY
                                        DECB    RCVCNT          ;DECREMENT THE COUNTER
                                        BGT     GETENT          ;HAS NOT EXPIRED
                                        ERRORK  <RECV TIMEOUT>  ;EXPIRED, PRINT ERROR MESSAGE
                                        JMP     GOABRT          ;SEND AN EOT AN ABORT
                                ;
                                ;
                                ;PARAMETER LIST FOR TIMED WAIT
                                ;
                                TTYWAT: .BYTE   377             ;OP CODE
                                        .BYTE   0               ;ID - LINE NUMBER
                                        .WORD   60              ;WAIT TIME - 1 SECOND (60 TICKS)
                                        .WORD   177777          ;USER PARAMETER WORD
                                ;
                                WAITNG: .WORD   0               ;FLAG TO INDICATE TIMER PENDING
                                ;
                                        .ENDC
```

```
                                        .GLOBL  COMERR          ;COMMAND ERROR
                                
002212                          BUFOVF: ERROR   <BUFF OVR>
002226  000167  177370                  JMP     GETENT
002232                          PRGERR: .IF DF  RSX
002232                                  CLOSE$  #FDBIN
002242                                  CLOSE$  #FDBOU0
002252                                  CLOSE$  #FDBOU1
                                        .ENDC
002262                                  FATAL   <CMTX ERR>
002276                          COMERR: .IF DF  RT11
                                        MOV     #STKSET,SP      ;RESET STACK(SWITCHES MAY BE THERE)
                                        .ENDC
002276                                  ERRORP  <ILL CMD>
002312  000167  177304                  JMP     CROUT
```

```
                                                .SBTTL  COMMAND COMPLETE FROM TTY
002316                                  COMCPL. .IF DF  RT11
                                                CMPB    -2(R1),#CR
                                                BNE     COMP2
                                                TSTB    -(R1)
                                        COMP2:  CLRB    -(R1)
                                                .CSISPC R2,#DEFEXT,R2   ;CALL THE CSI
                                                BCS     COMERR          ;COMMAND DECODER SYNTAX ERROR
                                                MOV     #FILDES,R1      ;ADDRESS OF CSI OUTPUT
                                                CMP     (R1),DEVRT      ;OUTPUT DEVICE RT?
                                                .ENDC
                                                .IF DF  RSX
002316  016701  176354                          MOV     IOST+2,R1
002322  001002                                 BNE     CT10
002324  000167  177266                          JMP     CROUTX
002330                                  CT10:   CSI$1   #FILDES,QDPBR+Q.IOPL,R1
002352  103412                                  BCS     BADCS           ;SYNTAX ERROR
002354                                          CSI$2   ,OUTPUT
002364  027067  000010  177064                  CMP     @C.DEVD+2(R0),DEVRT
                                                .ENDC
002372  001417                                  BEQ     TRNS4
002374  000167  000374                          JMP     TRYREC          ;MAYBE RECEIVE

                                                .IF DF  RSX
                                                .GLOBL  CSIERR
002400  004767  000002                  BADCS:  JSR     PC,CSIERR       ;PRINT STRING IN ERROR
002404  000734                                  BR      COMERR          ;TELL OPERATOR - COMMAND ERROR

002406  016767  004342  176244          CSIERR: MOV     FILDES+C.FILD+2,QDPBM+Q.IOPL    ;ADDRESS OF PAD STRING
002414  016767  004332  176240                  MOV     FILDES+C.FILD,QDPBM+Q.IOPL+2    ;LENGTH OF STRING
002422                                          DIR$    #QDPBM          ;WRITE THE MESSAGE
002430  000207                                  RTS     PC              ;RETURN TO PRINT REST OF MESSAGE
                                                .ENDC
```

```
                                                        .SBTTL  TRANSMIT COMMAND
                                                ;
002432  105767  177022          TRNS4:  TSTB    ONLINE          ;ONLINE?
002436  100402                          BMI     TRNS6           ;YES OK
002440  000167  177632                  JMP     COMERR          ;NO COMMAND ERROR
002444  012702  006504'          TRNS6:  MOV     #TRNSBLK,R2     ;TRANSMIT COMMAND PENDING?
002450  005732                          TST     @(R2)+  ;
002452  001410                          BEQ     MDTRY
002454                                  ERRORP  <TRNS PEND>
002470  000167  177126          TRNS7:  JMP     CROUT
002474                          MDTRY:          PUTMC   MDMPAR
002504  032767  000004  001464          BIT     #4,MDMPAR+4     ;IS DATA-LINK OK?
002512  001012                          BNE     TRNS5           ;YES - PROCEED
002514                                  ERROR   <MDM NOT RDY>
002530  005077  003750          TERRR:  CLR     @TRNSBLK        ;CANCEL TRANSMIT
002534  000167  177062                  JMP     CROUT
002540  010262  177776          TRNS5:  MOV     R2,-2(R2)       ;NO, SET POINTER TO FIRST FILEBLK
                                        .IF DF  RT11
                                        ADD     #36,R1          ;ADDRESS OF INPUT FILE BLOCK
                                TMOVE:  MOV     #4,R5           ;NUMBER OF WORDS PER FILE SPEC
                                                                ; NOT INCLUDING SWITCH WORD
                                10$:    MOV     (R1)+,(R2)+     ;STORE FILEBLK
                                        DEC     R5              ;ONE LESS  WORD TO MOVE
                                        BNE     10$             ;NOT DONE YET
                                        CLR     (R2)+           ;CLEAR SWITCH WORD
                                        CMP     R1,#FILDES+116  ;END OF INPUT FILE?
                                        BNE     TMOVE           ;NO, KEEP GOING
                                        CLR     (R2)+           ;MARK END OF TABLE WITH A ZERO
                                        JSR     PC,MSWIT        ;MOVE SWITCHES OFF STACK
                                        BCS     TERRR
                                        CLRB    SWRSL           ;NO ROOM FOR SEL RECV BUFFERS
                                         ENDC
002544  004767  000000G               JSR     PC,LUKUP        ;PARSE AND FIND FILES
002550  103747                        BCS     TRNS7           ;ERROR FROM LOOKUP
```

```
819                                             .SBTTL  MASTER BID
820
821 002552                                      .IIF DF RSX     CALL    PROMPT          ;DO PROMPT FOR RSX
822 002556  105067  176711                      CLRB    TDONE           ;RESET END FLAGS
823 002562  105067  176716                      CLRB    TRRTNF          ;PRINT PROMPT WHEN THROUGH
824 002566  105067  176677                      CLRB    ELATCH
825 002572  152767  000200  176711  BISIN:      BISB    #200,TXPEN      ;SET XMIT PENDING FLAG
826 002600  126727  176661  000001              CMPB    MODE,#1         ;RECEIVE ACTIVE?
827 002606  001466                              BEQ     GOTG            ;YES, WAIT FOR END OF TRANSMISSION
828                                             .IIF DF RT11    JSR     PC,DFNBUF       ;SET UP THE BUFFERS
829 002610  004767  000000G                     JSR     PC,FSTFIL       ;REOPEN FIRST FILE
830 002614  103016                              BCC     TROKT
831 002616                          BDFAIL:     ERROR   <XMIT OPEN ERR>
832 002632  105067  176653                      CLRB    TXPEN
833 002636  005077  003642                      CLR     @TRNSBLK
834 002642  004767  000000G                     JSR     PC,CLOSET       ;CLOSE THE FILE
835 002646  000167  176750                      JMP     CROUT
836 002652  132767  000040  176631  TROKT:      BITB    #40,TXPEN       ;TRANSMIT PENDING AFTER RVI?
837 002660  001403                              BEQ     10$             ;NO
838 002662  004767  000000G                     JSR     PC,POINT                ;POINT TO WHERE WE LEFT OFF IN FILE
839 002666  103753                              BCS     BDFAIL                  ;ERROR TRYING TO POSITION, GIVE UP
840 002670  004767  000000G          10$:       JSR     PC,FSTGET               ;DO THE INITIAL GET
841 002674  103750                              BCS     BDFAIL                  ;ERROR FROM GET, GIVE UP
842 002676  152767  000200  176556              BISB    #200,FLAGS      ; SELECT MASTER
843 002704  005067  001274                      CLR     CMPAR+4         ;FOR TRANSMISSION START
844 002710                                      PUTD    ENQPAR
845 002720  100423                              BMI     PUTERR          ;ERROR
846 002722  012767  040000  001202  JOINT:      MOV     #40000,ENBPAR+2 ;BUFFER ADDR IN PARAM LIST
847 002730  116767  176526  001174              MOVB    FLAGS,ENBPAR+2  ;SET CURRENT OPTIONS SELECTIONS
848 002736  156767  176517  001167              BISB    CODE,ENBPAR+3   ;SET CODE SELECTION
849 002744                                      PUTTC   ENBPAR  ;REQUEST CONTROL
850 002754  100405                              BMI     PUTERR
851 002756  152767  000200  176503              BISB    #200,INITF      ;SET INIT FLAG
852 002764  000167  176632          GOTG:       JMP     GETENT
853 002770  000167  177236          PUTERR:     JMP     PRGERR  ;COMTEX ERROR ;;** DEBUG **;;
```

```
                                        .SBTTL  RECEIVE TRANSFER COMMAND
                                ;
002774                          TRYREC: .IF DF  RT11
                                        CMP     FILDES+36,DEVRT
                                        BNE     VFCR            ;NOT DEVICE RT
                                        .ENDC
                                        .IF DF  RSX
002774                                  CSI$2   ,INPUT
003004  016000  000010                  MOV     C.DEVD+2(R0),R0
003010  122027  000122                  CMPB    (R0)+,#'R       ;DEVICE RT
003014  001030                          BNE     COMDD           ;NO
003016  121027  000124                  CMPB    (R0),#'T
003022  001025                          BNE     COMDD           ;NO
                                        .ENDC
003024  012702  006660'                 MOV     #RCVBLK+2,R2    ;RECEIVE COMMAND BLOCK
003030  122767  000001  176427          CMPB    #1,MODE         ;RECEIVE ACTIVE?
003036  001407                          BEQ     RPEND           ;YES
003040  010262  177776                  MOV     R2,-2(R2)       ;SET POINTER TO FIRST FILE
                                        .IF DF  RT11
                                        CLRB    RCHN            ;CLEAR RECEIVE CHAIN MODE
                                        CLRB    ZERFLG          ;CLEAR DELETED PREVIOUS FILE FLAG
                                RMOVE:  MOV     #5,R5           ;NUMBER OF WORDS PER FILE SPEC
                                                                ; NOT INCLUDING SWITCH WORD
                                10$:    MOV     (R1)+,(R2)+
                                        DEC     R5              ;ONE LESS WORD TO MOVE
                                        BNE     10$             ;NOT DONE YET
                                        CLR     (R2)+           ;CLREA OUT FILE SWITCH WORD
                                        CMP     R1,#FILDES+36   ;MOVED OUTPUT FILES?
                                        BNE     RMOVE           ;NO
                                        CLR     (R2)+           ;MARK END WITH 0
                                        JSR     PC,MSWIT        ;MOVE SWITCHES
                                        BCC     RPEND1
                                        CLR     @RCVBLK         ;ERROR ON SWITCHES
                                        BR      RPEND1          ;NO RECEIVE ALLOWED
                                        .ENDC
                                        .IF DF  RSX
003044  004767  000000G                 JSR     PC,PENTER       ;PARSE FILES AND SAVE FNB
003050  103410                          BCS     RPEND1          ;IF ERROR DON'T PRINT PROMPT
003052  000167  176540                  JMP     CROUTX          ;PRINT PROMPT AGAIN
                                        .ENDC
003056                          RPEND:  ERRORP  <RECV PEND>
003072  000167  176524          RPEND1: JMP     CROUT
```

```
                        .IF DF  RT11
                        .SBTTL  READ VERTICAL FORMAT RECORD
                        .GLOBL  DBLK,RESDNT
                ;
VFCR:           MOV     (SP)+,R2        ;NUMBER OF SWITCHES
                CMPB    R2,#1           ;ONE SWITCH?
                BNE     COMDD           ;NO, MORE OR LESS
                MOV     (SP)+,R2        ;GET SWITCH
                ADD     #36,R1          ;POINTER TO INPUT FILES
                CMPB    R2,#'V          ;VFC COMMAND?
                BNE     TRYDEF          ;NOT VFCR, TRY DEFAULT RECV
                TSTB    MODE            ;ARE WE IN CONTROL MODE
                BNE     CERROR          ;NO, COMMAND ERROR
                .DSTAT  #DBLK,R1
                BCS     CERROR          ;NO SUCH DEVICE
                TST     RESDNT          ;IS THE HANDLER RESIDENT
                BEQ     CERROR          ;NO- ERROR: FG CANNOT FETCH IT
                MOV     #LOKBLK,R0      ;SET UP FOR LOOKUP
                MOV     R1,2(R0)        ;MOVE DEVICE BLOCK
                .LOOKUP #2              ;DO A LOOKUP
                BCS     CERROR          ;NO SUCH FILE
                MOV     #REDBLK,R0      ;SET UP FOR READ
                MOV     #COLTT,4(R0)    ;MOVE BUFFER ADDRESS
                CLR     10(R0)          ;READ AND WAIT
                MOV     #400,6(R0)      ;WORDS TO READ
                MOV     #1,2(R0)        ;BLOCK 1
                .READ   #2              ;READ IT IN ON CHANNEL 2
                BCS     10$             ;A READ ERROR
                .CLOSE  #2              ;CLOSE THE FILE
                JMP     CROUT
10$:            .CLOSE  #2              ;CLOSE THE READ FILE
CERROR:         JMP     COMERR
TRYDEF:         CMPB    R2,#'D          ;DEF RECV COMMAND?
                BNE     CERROR          ;NO - ILLEGAL COMMAND
                CMPB    #1,MODE         ;RECEIVE ACTIVE?
                BEQ     RPEND           ;YES, NO GOOD
                MOV     R1,R2           ;SOURCE
                MOV     #5,R1           ;NUMBER OF WDS IN DEV BLK
                MOV     #DEFALT,R0      ;DESTINATION
                JSR     PC,XFRFNB       ;MOVE DEV BLK
                MOV     #17036,DFEXT    ;MOVE RAD50 ZERO INTO DEFAULT EXTENSION
                MOV     #17036,DFEXT+2  ;..
                CLR     @RCVBLK         ;CLEAR ANY PREVIOUS RECEIVE COMMANDS
                CLRB    ZERFLG          ;CLEAR "DELETED PREVIOUS FILE" FLAG
                CLRB    RCHN            ;CLEAR RECEIVE CHAIN MODE
                JMP     CROUT           ;ALL DONE
                 ENDC
```

```
                                        .SBTTL  TEST FOR COMMAND
003076                          COMDD:  .IF DF  RT11
                                        TST     R2              ;ANY SWITCHES?
                                        BNE     CERROR          ;AN ERROR
                                        MOV     FILDES+40,R0    ;FILE NAME
                                        .ENDC
                                        .IF DF  RSX
003076  016700  003636                  MOV     FILDES+C.CMLD+2,R0
003102  004767  000000G                 JSR     PC,RADCX
                                        .ENDC
003106  012705  004030'                 MOV     #COMATB,R5      ;COMD ACTION TBL ADDR IN R5
003112  020025                  10$:    CMP     R0,(R5)+        ;DO THREE CHARACTERS MATCH?
003114  001404                          BEQ     20$             ;YES, FOUND THE COMMAND
003116  005725                          TST     (R5)+           ;NO, ANY CMMDS LEFT TO TEST?
003120  001374                          BNE     10$             ;YES, KEEP LOOKING
003122  000167  177150                 JMP     COMERR          ;NO, ILLEGAL COMMAND
003126  105767  176333          20$     TSTB    MODE            ;TEST FOR MODE
003132  000175  000000                  JMP     @(R5)           ;GO TO THE PROPER ROUTINE

                                ;PRIMARY JUMP TABLE

003136  002232'                 JMPTBL: .WORD   PRGERR          ;R5=0  PROGRAM ERROR
003140  002212'                         .WORD   BUFOVF          ;R5=2  BUFFER OVERFLOW
003142  001622'                         .WORD   USRTIM          ;R5=4  USER TIMER EXPIRED
003144  000000G                         .WORD   INPCPL          ;R5=6  INPUT TYPE
003146  000000G                         .WORD   MSGDON          ;R5=8  OUTPUT TYPE
003150  001622'                         .WORD   GETENT          ;R5=10 STATUS TYPE
```

B	1	2	J	5	BISYNC	E	10	DPV112	M	14	RSXMC
C	1	BISYNC	K	5	BISYNC	F	10	DP11	N	14	COMTEX
D	1	BISYNC	L	5	BISYNC	G	10	DP11	B	15	COMTEX
E	1	BISYNC	M	5	BISYNC	H	10	DP11	C	15	COMTEX
F	1	BISYNC	N	5	BISYNC	I	10	DP11	D	15	RCSRSX
G	1	BISYNC	B	6	BISYNC	J	10	DUV209	E	15	RCSRSX
H	1	BISYNC	C	6	BISYNC	K	10	DUV209	F	15	RCSRSX
I	1	BISYNC	D	6	BISYNC	L	10	DUV209	G	15	RCSRSX
J	1	BISYNC	E	6	BISYNC	M	10	DUV209	H	15	RCSRSX
K	1	BISYNC	F	6	BISYNC	N	10	DUV209	I	15	RCSRSX
L	1	BISYNC	G	6	BISYNC	B	11	DUV209	J	15	RCSRSX
M	1	BISYNC	H	6	BISYNC	C	11	DUV209	K	15	RCSRSX
N	1	BISYNC	I	6	BISYNC	D	11	DUV209	L	15	RCSRSX
B	2	BISYNC	J	6	BISYNC	E	11	DUV209	M	15	RCSRSX
C	2	BISYNC	K	6	BISYNC	F	11	DUV209	N	15	RCSRSX
D	2	BISYNC	L	6	BISYNC	G	11	DUV209	B	16	RCSRSX
E	2	BISYNC	M	6	BISYNC	H	11	DUV209	C	16	RCSRSX
F	2	BISYNC	N	6	BISYNC	I	11	DUV209	D	16	RCSRSX
G	2	BISYNC	B	7	BISYNC	J	11	DUV209	E	16	RCSRSX
H	2	BISYNC	C	7	BISYNC	K	11	DUV209	F	16	RCSRSX
I	2	BISYNC	D	7	BISYNC	L	11	DUV209	G	16	RCSRSX
J	2	BISYNC	E	7	BISYNC	M	11	DUV209	H	16	RCSRSX
K	2	BISYNC	F	7	BISYNC	N	11	DUV209	I	16	RCSRSX
L	2	BISYNC	G	7	BISYNC	B	12	DUV209	J	16	RCSRSX
M	2	BISYNC	H	7	CLOSE	C	12	DUV209	K	16	RCSRSX
N	2	BISYNC	I	7	CLOSE	D	12	DUV209	L	16	RCSRSX
B	3	BISYNC	J	7	CLOSE	E	12	DUV209	M	16	RCSRSX
C	3	BISYNC	K	7	CLOSE	F	12	DUV209			
D	3	BISYNC	L	7	CLOSE	G	12	DUV209			
E	3	BISYNC	M	7	CLOSE	H	12	DUV209			
F	3	BISYNC	N	7	CLOSE	I	12	DUV209			
G	3	BISYNC	B	8	CLOSE	J	12	DU11			
H	3	BISYNC	C	8	DELETE	K	12	DU11			
I	3	BISYNC	D	8	DELETE	L	12	DU11			
J	3	BISYNC	E	8	DELETE	M	12	DU11			
K	3	BISYNC	F	8	DELETE	N	12	DU11			
L	3	BISYNC	G	8	DELETE	B	13	ERRHAN			
M	3	BISYNC	H	8	DPV112	C	13	ERRHAN			
N	3	BISYNC	I	8	DPV112	D	13	ERRHAN			
B	4	BISYNC	J	8	DPV112	E	13	ERRHAN			
C	4	BISYNC	K	8	DPV112	F	13	ERRHAN			
D	4	BISYNC	L	8	DPV112	G	13	ERRHAN			
E	4	BISYNC	M	8	DPV112	H	13	ERRHAN			
F	4	BISYNC	N	8	DPV112	I	13	ERRHAN			
G	4	BISYNC	B	9	DPV112	J	13	ERRHAN			
H	4	BISYNC	C	9	DPV112	K	13	ENTER			
I	4	BISYNC	D	9	DPV112	L	13	ENTER			
J	4	BISYNC	E	9	DPV112	M	13	ENTER			
K	4	BISYNC	F	9	DPV112	N	13	ENTER			
L	4	BISYNC	G	9	DPV112	B	14	ENTER			
M	4	BISYNC	H	9	DPV112	C	14	ENTER			
N	4	BISYNC	I	9	DPV112	D	14	FREE			
B	5	BISYNC	J	9	DPV112	E	14	FREE			
C	5	BISYNC	K	9	DPV112	F	14	FREE			
D	5	BISYNC	L	9	DPV112	G	14	FREE			
E	5	BISYNC	M	9	DPV112	H	14	FREE			
F	5	BISYNC	N	9	DPV112	I	14	FREE			
G	5	BISYNC	B	10	DPV112	J	14	FREE			
H	5	BISYNC	C	10	DPV112	K	14	FREE			
I	5	BISYNC	D	10	DPV112	L	14	COMTEX			

```
                                                        .SBTTL  SWITCH MODIFERS
                                                        .GLOBL  ONLHAN,TURNAR,TURN1
003152  001057                          ONLHAN: BNE     EBCHAN          ;NOT IN CONTROL MODE
003154  152767  000200  176276                BISB    #200,ONLINE     ;SET TO ONLINE
003162  016767  000064  000714                MOV     QONL,QSTAT      ;SET FOR QUERY
003170  116767  176266  000734                MOVB    FLAGS,ENBPAR+2  ;SET CURRENT OPTION SELECTIONS
003176  116767  176257  000727                MOVB    CODE,ENBPAR+3   ;SET CODE SELECTIONS
003204                                        PUTC    ENBPAR          ;
                                              .IF DF  RT11
                                              BPL     TURNAR          ;CONTINUE ON
                                              .ENDC
                                              .IF DF  RSX
003214  100006                              BPL     TURNON
                                              .ENDC
003216                                        FATAL   <HRD FAIL>
                                              .IF DF  RT11
                                        TURNAR:
                                        TURNON: CLRB    TRRTNF          ;CLEAR XMIT/RECV CODE RTN FLAG
                                              .ENDC
                                              .IF DF  RSX
003232                                  TURNON: CALL PROMPT             ;PROMPT THE OPERATOR
003236                                  TURNAR: :
                                              .ENDC
003236  004767  000374                  TURN1:  JSR     PC,CLRALL       ;CLEAR ALL FLAGS
003242  012767  002402  000712                MOV     #XSTEQ,STEQM
003250  012767  077404  000702                MOV     #XEOTS,EOTSEQ
003256  105767  176220                  BISIN2: TSTB    CMBFAV          ;ANY BUFFERS AVAILABLE
003262  001403                              BEQ     JOINTI
003264  004767  000000G                     JSR     PC,ASROUT       ;GO ASSIGN A BUFFER TO BISYNC TAP
003270  000772                              BR      BISIN2
003272  105767  176213                  JOINTI: TSTB    TXPEN           ;XMIT PENDING?
003276  100002                              BPL     JOINTX
003300  000167  177266                     JMP     BISIN
003304  000167  177412                  JOINTX: JMP     JOINT
003310  012446                          QCONT:  .RAD50  'CON'
```

```
1013                                                    .SBTTL  TRANSPARENT SWITCH
1014 003312  001100                            EBCHAN:  BNE     GOGETE
1015 003314  152767  000004  176140                    BISB    #4,FLAGS        ;CLEAR TRANSPARENCY FLAG
1016 003322  142767  000002  176132                    BICB    #2,FLAGS
1017 003330  000404                                    BR      COCOM
1018
1019 003332  001070                            EXBHAN:  BNE     GOGETE          ;NOT IN CONTROL MODE
1020 003336  152767  000006  176120                    BISB    #6,FLAGS        ;SET TRANSPARENCY FLAG
1021 003342  014567  000552                    COCOM:   MOV     -(R5),QCODE
1022 003346  000531                                    BR      COCOM1
1023
1024                                            ;
1025                                            ;;;;;;;;
1026                                            ;       BLOCKING MODE
1027                                            ;;;;
1028 003350  001061                            GNEMDE:  BNE     GOGETE          ;NOT IN CONTROL MODE
1029 003352  112767  000200  176103                    MOVB    #200,MCB
1030 003360  012767  000206  176132                    MOV     #134.,RECSIZ
1031                                                    .IF DF  RSX
1032 003366  012767  001000  174444                    MOV     #512.,F.RSIZ+FDBIN
1033 003374  112767  000001  174510                    MOVB    #1,F.MBCT+FDBIN ;SINGLE BUFFERED READS
1034                                                    .ENDC
1035 003402  000414                                    BR      COCOM2
1036                                            ;
1037 003404  001043                            RTMDE:   BNE     GOGETE          ;NOT IN CONTROL MODE
1038 003406  105067  176051                            CLRB    MCB             ;SET 2780 BLOCKING
1039                                                    .IF DF  RSX
1040 003412  012767  000120  174420                    MOV     #80.,F.RSIZ+FDBIN       ;MAX RECORD SIZE
1041 003420  012767  000001  174464                    MOV     #1,F.MBCT+FDBIN ;SINGLE BUFFERS FOR READING
1042                                                    .ENDC
1043 003426  012767  000120  176064                    MOV     #80.,RECSIZ
1044 003434  014567  000454                    COCOM2:  MOV     -(R5),QBLKM             ;BLOCKING MODE FOR QUERY TYPEOUT
1045 003440  000774                                    BR      COCOM1
```

```
1050                                            .SBTTL  TERM,DISCONNECT AND OFFLINE
1051
1052                                            .GLOBL  SNDEOT,OFFHAN,CLRMST,TERHAN
1053                                            .IF DF RT11
1054                                            .GLOBL  RCHN
1055                                            .ENDC
1056
1057                                    ;TERM HANDLER
1058 003442  105767  176012            TERHAN: TSTB    ONLINE
1059 003446  100022                            BPL     GOGETE
1060 003450  105767  176011                    TSTB    MODE            ;ARE WE IN CONTROL MODE?
1061 003454  001410                            BEQ     SNDEOT          ;YES, JUST SEND AN EOT
1062 003456  132767  000200  176024            BITB    #200,IOERR              ;HAVE WE ALREADY SET IO ERROR
1063 003464  001004                            BNE     SNDEOT          ;YES - GO SENT EOT - MUST BE HUNG
1064 003466  152767  000200  176014            BISB    #200,IOERR      ;NO, SIMULATE AN IO ERROR
1065 003474  000406                            BR      COCOM1          ;EOT WILL GET SENT INSTEAD OF ACK
1066 003476  000167  000000G           SNDEOT: JMP     GOABRT          ;SEND AN EOT IMMEDIATELY
1067                                    ;
1068
1069
1070
1071                                    ;DISCONNECT COMMAND HANDLER
1072 003502  105067  175776            DSCHAN: CLRB    TRRTNF          ;PRINT PROMPT WHEN DONE
1073 003506  105767  175746                    TSTB    ONLINE
1074 003512  100402                            BMI     DSCOUT
1075 003514  000167  176556            GOGETE: JMP     COMERR
1076 003520  105067  175760            DSCOUT: CLRB    TRRTNF          ;PRINT PROMPT
1077 003524                                    PUTD    DLEPAR          ;XMIT DLE EOT
1078 003534  152767  000200  175734            BISB    #200,EOTFLG     ;SET EOT FLAG
1079 003542  000167  000000G                   JMP     TER1            ;GO X-MIT DLE EOT
1080
1081
1082                                    ; OFFLINE COMMAND HANDLER
1083                                    ;(ALSO COMPLETES DISCONNECT COMMAND HANDLING)
1084
1085 003546  105067  000230            OFFHAN: CLRB    RCHN
1086 003552  105067  175702                    CLRB    ONLINE          ;ONLINE
1087 003556  105067  175727                    CLRB    TXPEN           ;CLEAR X-MIT PENDING
1088 003562  105067  175722                    CLRB    IOERR           ;CLEAR IOERROR FLAG
1089 003566                                    PUTTC   TERPAR          ;
1090 003576  016767  000226  000300    OFHAN1: MOV     COMATB,QSTAT    ;SET STATUS TO OFF
1091 003604  004767  000026                    JSR     PC,CLRALL       ;CLEAR ALL FLAGS
1092 003610  004767  000000G                   JSR     PC,CLOSET       ;CLOSE TRANSMIT FILES
1093 003614  004767  000000G                   JSR     PC,CLOSER       ;AND CLOSE RECEIVE FILES
1094                                            .IF DF  RSX
1095 003620  012777  177777  003030            MOV     #-1,@RCVBLK     ;NO MORE FILE INPUT
1096                                            .ENDC
1097                                            .IF DF  RT11
1098                                            CLR     @RCVBLK         ;NO MORE FILE INPUT
1099                                            .ENDC
1100 003626  005077  002652                    CLR     @TRNSBLK        ;OR OUTPUT
1101                                            .IF DF  RT11
1102                                    COCOM1: JMP     CROUT           ;GO PUT OFFLINE
1103                                            .ENDC
1104                                            .IF DF  RSX
1105 003632  000167  175760            COCOM1: JMP     CROUTX
1106                                            .ENDC
```

```
1107
1108                                        ; FLAG CLEARING ROUTINE FOR MODE TRANSITIONS
1109 003636  1050[illegible]  175623        CLRALL: CLRB    MODE            ;RETURN TO CONTROL MODE
1110 003642  005067  175646                 CLRMST: CLR     INPCNT
1111 003646  005067  175644                         CLR     NAKCNT
1112 003652  105067  175625                         CLRB    CMBFPN
1113 003656  105067  175610                         CLRB    ENDPEN          ;CLEAR END PENDING FLAG
1114 003662  105067  175606                         CLRB    EOTPEN          ;CLEAR EOT PENDING
1115 003666  105067  175604                         CLRB    EOTFLG          ;CLEAR EOT FLAG
1116 003672  142767  000200  175562                 BICB    #200,FLAGS      ;SET TO SLAVE MODE
1117 003700  016767  177404  000202                 MOV     QCONT,QMODE     ;SET UP QUERY TYPEOUT
1118 003706  000207                                 RTS     PC
```

```
1120                                            .IF DF RSX
1121                                            .SBTTL  CLEAR OUT THE TEMPORARY COMMON RECEIVE SWITCH AREA
1122                                    ;
1123                                    ; INPUTS:
1124                                    ;       NONE
1125                                    ;
1126                                            .GLOBL  CLRRSW
1127 003710  012700  000006          CLRRSW:  MOV     #RCVSW1/2,R0    ;# OF WORDS IN RECEIVE SWITCH AREA
1128 003714  012703  004014'                  MOV     #SWRB1,R3       ;FIRST SWITCH TO ZERO
1129 003720  005023                   10$:    CLR     (R3)+           ;CLEAR A WORD
1130 003722  005300                           DEC     R0              ;DECREMENT THE WORD COUNT
1131 003724  001375                           BNE     10$             ;MORE TO GO
1132 003726  012767  000206  000072           MOV     #134.,RVAL1     ;SET DEFAULT RECORD SIZE
1133 003734  012767  000000G 000060           MOV     #R.VAR,RTYP1    ;SET DEFAULT RECORD TYPE (VARIABLE)
1134 003742  012767  000000G 000054           MOV     #FD.CR,RATT1    ;SET DEFAULT RECORD ATTRIBUTES
1135 003750  000207                           RTS     PC              ;ALL DONE
1136                                            .ENDC
```

```
                                        .SBTTL  TABLE FOR WRITE
                                ;
                                ;       TWO FILES MAY BE OPEN FOR WRITE
                                ;       AT THE SAME TIME (SELECTIVE RECEIVE)
                                ;
                                        .IF DF RSX
003752  000176'                 CHANL:: .WORD   FDBOU0  ;FDB BLOCK ADDRESS
003754  000000                  COLP1:: .WORD   0       ;FOR VERTICAL FORMS CONTROL
003756  000000                  RECS1:: .WORD   0       ;FOR FIXED LENGTH RECORDS
003760  000336'                 CHANL2:: .WORD  FDBOU1  ;SECOND FDB ADDRESS
003762  000000                  COLP2:: .WORD   0       ;FOR VERTICAL FORMS CONTROL
003764  000000                  RECS2:: .WORD   0       ;FOR FIXED LENGHTH RECORDS
003766  000000                  COLAD:: .WORD   0
003770  000000                  CHARL:: .WORD   0
                                        .ENDC
                                ;
                                ;
                                        .IF DF RT11
                                        .GLOBL  ICHANL,IPBUFF,ICBUFF,COLPNT,IWREND
                                        .GLOBL  IWRCP,IOUBLK,CHANL,PBUFF,CBUFF,COLP1
                                        .GLOBL  WREND,OUBLK,CHANL2,PBUFF1,CBUFF1,COLP2
                                        .GLOBL  WRTCP
                                        .GLOBL  WREND1,OUBLK1.COLAD,CHARL,CHARF
                                ;
                                ICHANL=0
                                IPBUFF=2                        ;PENDING BUFFER
                                ICBUFF=4                        ;CURRENT BUFFER
                                COLPNT=6
                                IWREND=10
                                IWRCP=12
                                IOUBLK=14
                                CHANL:  .WORD   12
                                PBUFF:  .WORD   BUF8
                                CBUFF:  .WORD   BUFA
                                COLP1:  .WORD   0
                                WREND:  .WORD   BUFA+1004
                                        .WORD   WRTCP
                                OUBLK:  .WORD   0
                                CHANL2: .WORD   13
                                PBUFF1: .WORD   0
                                CBUFF1: .WORD   0
                                COLP2:  .WORD   0
                                WREND1: .WORD   0
                                        .WORD   WRTCP
                                OUBLK1: .WORD   0
                                COLAD:  .WORD   0
                                CHARL:  .WORD   0
                                CHARF:  .WORD   0
                                        .ENDC
```

```
1188                                            .IF DF RSX
1189                                            .SBTTL  SWITCH TABLE AND ROUTINES
1190
1191 003772  000000                    SWTBN:: .WORD   0               ;TRANSMIT BINARY SWITCH
1192 003774  000000                    SWREN:: .WORD   0               ;END SWITCH
1193         000004                            XMTSWT==.-SWTBN         ;NUMBER OF BYTES OF TRANSMIT COMMAND SWITCHES
1194 003776  000000                    SWRBN:: .WORD   0               ;RECEIVE BINARY SWITCH
1195 004000  000000                    SWRSL:: .WORD   0               ;SELECTIVE RECEIVE SWITCH
1196 004002  000000                    RCHN::  .WORD   0               ;RECEIVE CHAIN MODE SWITCH
1197 004004  000000                    RTYP::  .WORD   0               ;RECORD TYPE FOR FDB
1198 004006  000000                    RATT::  .WORD   0               ;RECORD ATTRIBUTES FOR FDB
1199 004010  000000                    RVAL::  .WORD   0               ;RECORD SIZE FOR FDB
1200         000014                            RCVSWT==.-SWRBN         ;NUMBER OF BYTES OF RECEIVE COMMAND SWITCHES
1201 004012  000000                    SWRDF:: .WORD   0               ;CHANGE DEFAULT RECEIVE FILE
1202                                                                    ;(DON'T INCLUDE AS ONE OF THE "RCVSWT"
1203                                                                    ;SWITCHES SINCE THE FILE DOES NOT
1204                                                                    ;GET STORED INTO RCVBLK)
1205                                    ; TEMPORARY SWITCHES TO AVOID A /DF WIPING OUT THE
1206                                    ; SWITCHES ON THE OTHER RECEIVE FILE
1207 004014  000000                    SWRB1:: .WORD   0
1208 004016  000000                    SWRS1:: .WORD   0
1209 004020  000000                    RCHN1:: .WORD   0
1210 004022  000000                    RTYP1:: .WORD   0
1211 004024  000000                    RATT1:: .WORD   0
1212 004026  000000                    RVAL1:: .WORD   0
1213         000014                    RCVSW1==.-SWRB1
1214                                            .ENDC
1215
1216                                            .IF DF  RT11
1217                                            .GLOBL  SWRSL,SWREN,SWRBN,RCHN
1218                                            .IF DF LSI11
1219                                            .GLOBL  SWRSL1
1220                                            .ENDC
1221
1222                                    RCHN:   .WORD   0               ;RECEIVE CHAIN MODE FLAG
1223                                    SWRBN:  .BYTE   0               ;BINARY MODE FLAG
1224                                    SWRSL:  .BYTE   0               ;SELECTIVE RECEIVE FLAG
1225                                    SWREN:  .BYTE   0               ;END MODE FLAG
1226                                            .IF DF LSI11
1227                                    SWRSL1: .BYTE   0               ;SEL REC SAVE FLAG
1228                                            .ENDC
1229
1230                                            .EVEN
1231                                            .ENDC
```

```
                                        ;SWITCH MODIFIER COMMAND ACTION TABLE

                                                .GLOBL  XFEREN
                                        ;
004030  057266                  COMATB:       .RAD50  'OFF'
004032  003546'                               .WORD   OFFHAN
004034  057774                  QONL:         .RAD50  'ONL'
004036  003152'                               .WORD   ONLHAN
004040  021402                                .RAD50  'EXB'
004042  003332'                               .WORD   EXBHAN
004044  017623                                .RAD50  'EBC'           ;EXBCID CODE
004046  003312'                               .WORD   EBCHAN

                                        ;CONTROL COMMAND ACTION TABLE
004050  076732                                .RAD50  'TER'
004052  003442'                               .WORD   TERHAN
004054  015173                                .RAD50  'DIS'           ;DISCONNECT
004056  003502'                               .WORD   DSCHAN
                                        ;RECORD BLOCKING SETTING
004060  026226                                .RAD50  'GEN'
004062  003350'                               .WORD   GNEMDE
004064  071025                                .RAD50  'RJE'
004066  003404'                               .WORD   RTMDE

                                        ;MISCELLANEOUS COMMAND ACTION TABLE

004070  066615                                .RAD50  'QUE'           ;QUERY
004072  000000G                               .WORD   QUEHAN
004074  021042                                .RAD50  'ERR'
004076  000000G                               .WORD   ERRHAN
                                              .IF DF  RT11
                                              .RAD50  'EXT'           ;EXIT COMMAND
                                              .WORD   DONE
                                              .ENDC
004100  100000                  XFEREN:     .WORD     100000          ;END OF TABLE MARKER
004102  000000                              .WORD     0               ;THIS TOO
```

```
1273                                    .SBTTL  QUERY STATUS STORAGE AREA
1274
1275
1276
1277                                    ;QUERY MESSAGE STORAGE (DEFAULTS ARE ALREADY IN MESSAGE)
1278                                    ; THE VARIOUS COMMANDS CHANGE THE MESSAGE IN APPROPRIATE PLACES
1279
1280                                            .GLOBL  QSTAT,QMODE,QBLKM,QEND,QLENTH,QLNGTH
1281                                    ;
1282 004104  057266                    QSTAT:  .RAD50  'OFF'           ;
1283 004106  000000                            .RAD50  '   '
1284 004110  012446                    QMODE:  .RAD50  'CON'
1285 004112  000000                            .RAD50  '   '
1286 004114  071025                    QBLKM:  .RAD50  'RJE'
1287 004116  000000                            .RAD50  '   '
1288 004120  017623                    QCODE:  .RAD50  'EBC'
1289 004122  000000                            .RAD50  '   '
1290 004124                            QEND:
1291         000030                             QLNGTH=<QEND-QSTAT>/2*3
1292         000020                             QLENTH=QEND-QSTAT
```

```
                                                .SBTTL  PARAMETERS
                                                .GLOBL  ITBM,ENBPAR,BISPAR,RESP,EOTSEQ,CMPAR
                                        ;
004124  077405                  ENQM:   .WORD   XCEP

004126     037                  ITBM:   .BYTE   XITB
004127     000                          .BYTE   0

004130     000                  ENBPAR: .BYTE   BASELN
004131     002                          .BYTE   2
004132  000000                          .WORD   0
004134  000000                          .WORD   0

004136     000                  DLEPAR: .BYTE   BASELN
004137     100                          .BYTE   100
004140  004164'                         .WORD   DLEEOT
004142  000002                          .WORD   2

004144     000                  ENQPAR: .BYTE   BASELN
004145     100                          .BYTE   100
004146  004124'                         .WORD   ENQM
004150  000001                          .WORD   1


004152     000                  BISPAR: .BYTE   BASELN
004153     003                  RESP:   .BYTE   03
004154  000000                          .WORD   0
004156  000206                          .WORD   LPSIZE

004160  077404                  EOTSEQ: .WORD   XEOTS
004162  002402                  STEQM:  .WORD   XSTEQ

004164  002020                  DLEEOT: .WORD   XDEOT
004166     177                          .BYTE   XPAD
004167     000                          .BYTE   0
004170  000003                          .WORD   3

004172     000                  MDMPAR: .BYTE   BASELN
004173     201                          .BYTE   201
004174  100000                          .WORD   100000
004176  000000                          .WORD   0

004200     000                  CMPAR:  .BYTE   BASELN
004201     000                          .BYTE   0
004202  000000                          .WORD   0
004204  000000                          .WORD   0

004206     000                  TERPAR: .BYTE   BASELN
004207     002                          .BYTE   2
```

```
1351 004210  140000                          .WORD   140000
1352 004212  000000                          .WORD   0
1353
1354                                         .IF DF RT11
1355                                    ;
1356                                    ; EXTRA QUEUE ELEMENT STORAGE
1357                                         .GLOBL  QLSTRT
1358                                    ;
1359                                    QLSTRT: .BLKW   70.             ;10., 7-WORD QUEUE ELEMENTS
1360                                         .ENDC
```

```
                                .SBTTL  COMMUNICATION BUFFER STORAGE

                                .GLOBL  CMBUF1,CMBUFE,HFCREC,TRNSBLK
                                .GLOBL  RCVBLK,FILDES,KBIBUF

004214                  CMBUF1:                         ;START OF COMM BUFFERS

                        CMBUF2=CMBUF1+LPSIZE
                        CMBUF3=CMBUF2+LPSIZE
                        CMBUF4=CMBUF3+LPSIZE
                        CMBUF5=CMBUF4+LPSIZE
                        CMBUF6=CMBUF5+LPSIZE
                        CMBUF7=CMBUF6+LPSIZE
                        CMBUFE=7*LPSIZE+CMBUF1          ;SPACE FOR 7 CR BUFFERS OF
                                                        ;80 BYTES OR 7 LP BUFFERS OF
                                                        ;"LPSIZE" BYTES

                                .=CMBUFE                ;END OF COMM BUFFER REGION
006066  006066          HFCREC: .BLKB   LPSIZE          ;HFC RECORD
006274  377                     .BYTE   377             ;MARKS END OF HORIZONTAL FORMAT RECORD
                                .EVEN

                                .IF DF  RSX
006276                  EXPREC: .BLKB   LPSIZE
                                .ENDC
006504  006506'         TRNSBLK: .WORD  TRNSBLK+2       ;POINTER TO BLOCK WHERE XMIT CMMDS ARE PARSED
006506  000000                 .WORD   0               ;LAST PARSED XMIT FILE IS MARKED BY
                                                        ; A ZERO IN THE FIRST WORD FOLLOWING IT

                                .IF DF  RT11
                                .BLKW   4
                                .BLKW   5
                                .BLKW   5
                                .BLKW   5
                                .BLKW   5
                                .WORD   0               ;END OF FILE BLOCK
                                .ENDC
                                .IF DF  RSX
                                .BLKB   S.FNB+XMTSWT-2
000002                          .REPT   NTRNSF
                                .BLKB   S.FNB
                                .BLKB   XMTSWT
                                .ENDM
006654  000000          TRNSE:: .WORD   0               ;LAST LOC OF XMIT FILE STORAGE REGION
                                ENDC
006656  006660'         RCVBLK: .WORD   RCVBLK+2        ;POINTER TO NEXT RECEIVE COMMAND
                                .IF DF  RSX
006660  177777                 .WORD   -1              ;LAST PARSE RECV FILE IS MARKED BY
                                                        ; A MINUS ONE IN THE FIRST WORD FOLLOWING IT
                                .ENDC

                                .IF DF  RT11
                                .WORD   0               ;A ZERO MARKS THE LAST FILE ENTRY FOR RT11
                                .BLKW   5
                                .BLKW   6
                                .BLKW   6
                                .WORD   0
```

```
                                        .ENDC
                                        .IF DF  RSX
                                        .BLKB   S.FNB+RCVSWT-2
000000                                  .REPT   NRCVSF
                                        .BLKB   S.FNB
                                        .BLKB   RCVSWT
                                        .ENDM
006732  177777                  RCVSE:: .WORD   -1              ;LAST LOC OF RECV FILE STORAGE REGION
                                        .ENDC
                                ;CONTROL BLOCK FOR COMMAND STRING INTERPRETER (CSI)
                                        .IF DF  RT11
                                FILDES:
                                KBIBUF: .BLKB   80.

                                        .GLOBL  RTEND

                                RTEND:  .WORD   0       ;EOF FLAG FOR READ


                                        .SBTTL  READ PARAMETER BLOCK

                                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                ; PARAMETER BLOCK FOR .READ EMT
                                        .GLOBL  REDBLK

                                REDBLK: .BYTE   0               ;CHANNEL
                                        .BYTE   10              ;FUNCTION (READ)
                                        .WORD   0               ;BLOCK # FOR READ
                                        .WORD   0               ;BUFFER FOR TRANSFER
                                        .WORD   400             ;WORD COUNT
                                        .WORD   0               ;TYPE OF READ
                                                                ;0 - READ
                                                                ;1 - READW
                                                                ;OTHER IS COMPLETION ROUTINE
                                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

                                        .GLOBL  BLKSAV,WRDSAV


                                BLKSAV: .WORD   0               ;STORAGE FOR CURRENT BLOCK FOR XMIT (RVI)
                                WRDSAV: .WORD   0               ;STORAGE FOR CURRENT WORD TO XMIT

                                        .GLOBL  SAVEST,SAVBLK,LOKBLK,DBLK,RESDNT

                                SAVEST: .WORD   0
                                ;
                                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                ;LOOKUP PARAMETER BLOCK FOR EMT CALL .LOOKUP
                                LOKBLK: .BYTE   0               ;CHANNEL
                                        .BYTE   1               ;FUNCTION CODE
                                        .WORD   0               ;DEVBLK
                                        .WORD   0               ;COUNT
                                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                ;
                                ;
                                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                ; SAVEST PARAMETER BLOCK FOR EMT CALL .SAVESTATUS
```

```
                                SAVBLK: .BYTE   11              ;CHANNEL
                                        .BYTE   5               ;FUNCTION CODE
                                        .WORD   0               ;POINTER TO SAVEST BLOCK
                                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                ;
                                ;
                                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                ;DSTATUS PARAMETER BLOCK FOR EMT CALL .DSTAT
                                ;
                                DBLK:   .WORD 0,0
                                RESDNT: .WORD 0,0               ;HANDLER ENTRY POINT
                                ;
                                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                        .ENDC
                                        .IF DF  RSX
006734                          FILDES: .BLKB   C.SIZE
                                        .EVEN
000010                          KBIBUF: .BLKB   80.
                                        .ENDC
```

```
1498                                                    .IF DF  RSX
1499 000247                                             .PSECT  MESAG2
1500 000247     122     112     105  PRMTA:  .ASCII  "RJE>**"
     000252     076     052     052
1501          000255'                                   PREND=.
1502          000005                                    PRMTL=PREND-PRMTA-1
1503                                                    .EVEN
1504 007130                                             .PSECT  MAIN
1505                                                    .ENDC
1506          001206'                                   .END LININT
```

RCSRSXM.F001 MACRO M1104 06-DEC-77 08:21 PAGE 52-1
SYMBOL TABLE

```
ACK    = 000040 G      COMDD   003076R     003 D$$YNC= 000000        F.FACC= 000043      JOINT   002722R     003
ACKPEN  001470RG   003 COMERR  002276RG    003 D$$YNM= 000000        F.FFBY= 000014      JOINTI  003272R     003
ASROUT= ****** G       CR     = 000015        EBCHAN  003312R    003 F.FNAM= 000110      JOINTX  003304R     003
ASSGNX  001505RG   003 CRBASX  001524RG    003 ELATCH  001471RG   003 F.FNB = 000102      KBIBUF  007010RG    003
A$$CHK= 000000         CRBFAV  001501RG    003 EMTERR  001320R    003 F.FTYP= 000116      K$$CNT= 172542
A$$CPS= 000000         CROUT   001622RG    003 EMTGD   001246R    003 F.FVER= 000120      K$$CSR= 172540
A$$NSI= 000000         CROUTX  001616RG    003 EMTGET  001256RG   003 F.HIBK= 000004      K$$LDC= 000024
A$$PRI= 000000         CSIERR  002406RG    003 EMTINT  001272RG   003 F.LUN = 000042      X$$TPS= 000764
A$$TRP= 000000         CS.DIF= 000002          EMTPUT  001234RG   003 F.MBCT= 000054      LD$DB = 000000
BADCS   002400R    003 CS.DVF= 000004          EMTTIM  001306RG   003 F.MBC1= 000055      LD$DK = 000000
BASELN= 000000 G       CS.EQU= 000040          ENBPAR  004130RG   003 F.MBFG= 000056      LD$DS = 000000
BDFAIL  002616R    003 CS.INP= 000001          ENDPEN  001472RG   003 F.NRBD= 000024      LD$DT = 000000
BINEOT= 177467 G       CS.MOR= 000020          ENQM    004124R    003 F.NREC= 000030      LD$LP = 000000
BINESC= 000047 G       CS.NMF= 000001          ENQPAR  004144R    003 F.OVBS= 000030      LD$MM = 000000
BINITB= 000037 G       CS.OUT= 000002          ENTRY   001764RG   003 F.RACC= 000016      LD$PP = 000000
BINTTD= 026402 G       CS.WLD= 000010          EOTFLG  001476RG   003 F.RATT= 000001      LD$PR = 000000
BIN4   = 000364 G      CTEMT  = ****** GX      EOTPEN  001474RG   003 F.RCNM= 000034      LF     = 000012
BISIN   002572R    003 CTIO    002330R     003 EOTSEQ  004160RG   003 F.RCTL= 000017      LININT  001206RG    003
BISIN2  003256R    003 C$$CKP= 000000          ERRHAN= ****** G       F.RSIZ= 000002      LPSIZE= 000206 G
BISPAR  004152RG   003 C$$ORE= 004000          ESC    = 000033 G      F.RTYP= 000000      LUKUP  = ****** G
BPTINS  001362R    003 C$$RSH= 177564          ETXFLG  001475RG   003 F.SEQN= 000100      L$$ASG= 000000
BUFOVF  002212R    003 C.BODE= ****** GX       EXBHAN  003332R    003 F.SPDV= 000072      L$$DRV= 000000
CHANL   003752RG   003 C.CMLD= 000002          EXPREC  006276R    003 F.SPUN= 000074      L$$P11= 000001
CHANL2  003760RG   003 C.DEVD= 000006          E$$DVC= 000000          F.STBK= 000036      L$$11R= 000000
CHARL   003770RG   003 C.DIRD= 000012          E$$NSI= 000000          F.UNIT= 000136      MCB     001463RG    003
CLOSER= ****** G       C.DSDS= 000006          E$$PER= 000000          F.URBD= 000020      MDMPAR  004172R     003
CLOSET= ****** G       C.FILD= 000016          E$$XPR= 000000          F.VBN = 000064      MDTRY   002474R     003
CLRALL  003636R    003 C.MKW1= 000024          FDBIN   000036RG   003 F.VBSZ= 000060      MEMPRO  001346R     003
CLRMST  003642RG   003 C.MKW2= 000026          FDBOU0  000176RG   003 GETENT  001622RG   003 MODE    001465RG    003
CLRRSW  003710RG   003 C.SGET= ****** GX       FDBOU1  000336RG   003 GETEXP  001774R    003 MSGDON= ****** G
CMBFAV  001502RG   003 C.SIZE= 000054          FD.CR  = ****** GX      GETO1   001654R    003 M$$CRB= 000124
CMBFPN  001503RG   003 C.SPS  = ****** G       FD.PLC= ****** GX       GNEMDE  003350R    003 M$$CRX= 000000
CMBLST  001526RG   003 C.SPUT= ****** GX       FILDES  006734RG   003 GOABRT= ****** G         M$$EXT= 000000
CMBMAX= 000016 G       C.SREG= ****** GX       FLAGS   001462RG   003 GOGETE  003514R    003 M$$FCS= 000000
CMBUFE= 006066RG   003 C.SRES= ****** GX       FSTFIL= ****** G        GOTG    002764R    003 M$$MGE= 000000
CMBUF1  004214RG   003 C.STAB= ****** GX       FSTGET= ****** G        GOTGET  001616RG   003 M$$MUP= 000000
CMBUF2= 004422R    003 C.STAT= 000001          F$$LVL= 000001          G$$TPP= 000000          M$$OVR= 000000
CMBUF3= 004630R    003 C.STIN= ****** GX       F.ACTL= 000076          G$$TTK= 000000          NAK    = 000100 G
CMBUF4= 005036R    003 C.SWAD= 000022          F.ALOC= 000040          HFCACT  001466RG   003 NAKCNT  001516RG    003
CMBUF5= 005244R    003 C.TYPR= 000000          F.BBFS= 000062          HFCREC  006066RG   003 NRCVSF= 000000
CMBUF6= 005452R    003 DEVRT   001456RG    003 F.BDB  = 000070          IE.EOF= ****** GX       NTRNSF= 000002
CMBUF7= 005660R    003 DFBLK   000000RG    003 F.BGBC= 000057          ILLINS  001426R    003 N$$LDV= 000001
CMCLST  001562RG   003 DFNLP   000476RG    003 F.BKDN= 000026          INCRCV  001477RG   003 N$$MOV= 000021
CMFLST  001600RG   003 DIE     002206RG    003 F.BKDS= 000020          INITAL= ****** G        N.DID  = 000024
CMPAR   004200RG   003 DLEEOT  004164R     003 F.BKEF= 000050          INITF   001467RG   003 N.DVNM= 000032
CMPLST  001544RG   003 DLEPAR  004136R     003 F.BKP1= 000051          INPCNT  001514RG   003 N.FID  = 000000
COCOM   003342R    003 DONE   = ****** G       F.BKST= 000024          INPCPL= ****** G        N.FNAM= 000006
COCOM1  003632R    003 DSCHAN  003502R     003 F.BKVB= 000064          IOERR   001510RG   003 N.FTYP= 000014
COCOM2  003434R    003 DSCOUT  003520R     003 F.CHR  = 000075          IOST    000674R    003 N.FVER= 000016
CODE    001461RG   003 DSW     000000R         F.CNTG= 000034          IOTINS  001376R    003 N.NEXT= 000022
COLAD   003766RG   003 D$$BUG= 177514          F.DFNB= 000046          IO.KIL= ****** GX       N.STAT= 000020
COLP1   003754RG   003 D$$H11= 000002          F.DSPT= 000044          IO.RLB= ****** GX       N.UNIT= 000034
COLP2   003762RG   003 D$$IAG= 000000          F.DVNM= 000134          IO.WLB= ****** GX       ODDADD  001332R     003
COLTT  = ****** G       D$$ISK= 000000          F.EFBK= 000010          ITBM    004126RG   003 OFFHAN  003546RG    003
COMATB  004030R    003 D$$L11= 000001          F.EFN  = 000050          I$$RAR= 000000          OFHAN1  003576R     003
COMAV  = 000007 G       D$$M11= 000001          F.EOBB= 000032          I$$RDN= 000000          ONLHAN  003152RG    003
COMCPL  002316R    003 D$$SHF= 000000          F.ERR  = 000052          JMPTBL  003136R    003 ONLINE  001460RG    003
```

SYMBOL TABLE

```
OPNFLG  001513RG    003 RCVTIM= 000177 G        S.FNTY= 000004          TRRTNF  001504RG    003 T.RCVL  000012
PAR$$$= 000027          RCVTMT  001200RG    003 S.FTYP= 000002          TRYREC  002774R     003 T.RRFL  000064
PENTER= ****** GX       RECSIZ  001520RG    003 S.NFEN= 000020          TS.BLK= 170700          T.SRCT  000063
POINT = ****** G        RECS1   003756RG    003 S.VTTA= 000002          TS.CKP= 000200          T.STAT  000032
POINT1  000700RG    003 RECS2   003764RG    003 S.VTTL= 000004          TS.CKR= 000100          T.ST2   000034
POINT2  000702RG    003 RESP    004153RG    003 TABSST  000714RG    003 TS.EXE= 100000          T.ST3   000036
POINT3  000704RG    003 RPEND   003056R     003 TAGE1   000010R     005 TS.MSG= 020000          T.TCBL  000030
PREND = 000255R     005 RPEND1  003072R     003 TAGE10  000103R     005 TS.NRP= 010000          T.UCB   000026
PRGERR  002232R     003 RSVDIN  001412R     003 TAGE11  000111R     005 TS.OUT= 000400          T2.ABO= 000100
PRINTX  001507RG    003 RSX   = 000001          TAGE12  000136R     005 TS.RDN= 040000          T2.AST= 100000
PRMTA   000247R     005 RSX11M= 000001          TAGE13  000146R     005 TS.RUN= 004000          T2.BFX= 004000
PRMTL = 000005          RTMDE   003404R     003 TAGE14  000156R     005 TTERFN= 000020          T2.CAF= 000400
PROMPT  002142RG    003 RTYP    004004RG    003 TAGE15  000165R     005 TTKLFN= 000004          T2.CHK= 020000
PS    = 177776          RTYP1   004022RG    003 TAGE16  000176R     005 TTPRFN= 000002          T2.CKD= 010000
PUTERR  002770R     003 RVAL    004010RG    003 TAGE17  000211R     005 TTRDFN= 000010          T2.DST= 040000
P$$D70= 000000          RVAL1   004026RG    003 TAGE2   000020R     005 TURNAR  003236RG    003 T2.FXD= 002000
P$$GMX= 000000          RVIPEN  001500RG    003 TAGE20  000226R     005 TURNON  003232R     003 T2.HLT= 000200
P$$LAS= 000000          R$$DER= 000000          TAGE21  000237R     005 TURN1   003236RG    003 T2.SPN= 000004
P$$P11= 000001          R$$EXV= 000000          TAGE22  000247R     005 TVAB    001464RG    003 T2.STP= 000020
P$$P45= 000000          R$$JP0= 000000          TAGE3   000027R     005 TXPEN   001511RG    003 T2.TIO= 001000
P$$RFL= 000000          R$$JP1= 000001          TAGE4   000036R     005 TYPMSG  002112RG    003 T2.WFR= 000001
P$$RTY= 000000          R$$JS1= 000001          TAGE5   000046R     005 TYPMS1  002120RG    003 T3.ACP= 100000
P$$R11= 000001          R$$KWC= 000000          TAGE6   000055R     005 T$$ACR= 000000          T3.CAL= 000100
P$$SRF= 000000          R$$K11= 000001          TAGE7   000065R     005 T$$BTW= 000000          T3.CLI= 001000
QBLKM   004114RG    003 R$$NDC= 000003          TAGNUM= 000022          T$$BUF= 000000          T3.MCR= 004000
QCODE   004120R     003 R$$NDH= 000226          TAG1    000000R     005 T$$CCA= 000000          T3.NET= 000020
QCONT   003310R     003 R$$NDL= 000001          TAG10   000065R     005 T$$CCO= 000000          T3.NSD= 000200
QDPBK   000614R     003 R$$SND= 000000          TAG11   000103R     005 T$$CTR= 000000          T3.PMD= 040000
QDPBM   000644RG    003 R$$11M= 000000          TAG12   000111R     005 T$$C11= 000001          T3.PRV= 010000
QDPBR   000564R     003 R.VAR = ****** GX       TAG13   000136R     005 T$$GMC= 000000          T3.REM= 020000
QDPBW   000534R     003 SIBEFN= 000040          TAG14   000146R     005 T$$GTS= 000000          T3.ROV= 000040
QEND    004124RG    003 SLEEP   002060R     003 TAG15   000156R     005 T$$J16= 000001          T3.RST= 000400
QLENTH= 000020 G        SNDEOT  003476RG    003 TAG16   000165R     005 T$$KMG= 000000          T3.SLV= 002000
QLNGTH= 000030 G        SSTVEC  000706R     003 TAG17   000176R     005 T$$LWC= 000000          UBFCNT  001204RG    003
QMODE   004110RG    003 STEQM   004162R     003 TAG2    000010R     005 T$$MAN= 006307          UIBEFN= 000001
QONL    004034R     003 STKLMT  000732R     003 TAG20   000211R     005 T$$RNE= 000000          USRTIM  001622RG    003
QSTAT   004104RG    003 STKSET  001174RG    003 TAG21   000226R     005 T$$RPR= 000000          U$$MHI= 000040
QUEHAN= ****** G        STOREX  001506RG    003 TAG22   000237R     005 T$$RST= 000000          U$$MLO= 120000
Q$$OPT= 000017          SWRBN   003776RG    003 TAG3    000020R     005 T$$RUB= 000000          U$$MRN= 170324
Q.IOAE= 000012          SWRB1   004014RG    003 TAG4    000027R     005 T$$SYN= 000000          V$$CTR= 000400
Q.IOEF= 000006          SWRDF   004012RG    003 TAG5    000036R     005 T$$TRW= 000000          XCEP  = 077405
Q.IOFN= 000002          SWREN   003774RG    003 TAG6    000046R     005 T$$VBF= 000000          XDEOT = 002020
Q.IOLU= 000004          SWRSL   004000RG    003 TAG7    000055R     005 T.ACTL  000052          XEOTS = 077404 G
Q.IOPL= 000014          SWRSUP  001522RG    003 TASKAD= ****** GX       T.ASTL  000016          XFEREN  004100RG    003
Q.IOPR= 000007          SWRS1   004016RG    003 TDONE   001473RG    003 T.ATT   000054          XITB  = 000037 G
Q.IOSB= 000010          SWTBN   003772RG    003 TERHAN  003442RG    003 T.CPCB  000004          XMTERR  001176RG    003
RADCX = ****** GX       SYSERR  002076R     003 TERPAR  004206R     003 T.DPRI  000040          XMTSWT= 000004 G
RATT    004006RG    003 S$$WPC= 000036          TERR    002530R     003 T.EFLG  000022          XMTTMT  001202RG    003
RATT1   004024RG    003 S$$WPR= 000005              TER1  = ****** G      T.IOC   000003          XPAD  = 000177
RCHN    004002RG    003 S$$WRG= 000000          TRNSBL  006504RG    003 T.LBN   000041          XSTEQ = 002402 G
RCHN1   004020RG    003 S$$YSZ= 024000          TRNSE   006654RG    003 T.LDV   000044          X$$DBT= 000000
RCVBLK  006656RG    003 S.BFHD= 000020          TRNS4   002432R     003 T.LNK   000000          $$$ARG= 000003
RCVCNT  001512RG    003 S.FATT= 000016          TRNS5   002540R     003 T.MXSZ  000050          $$$OST= 000006
RCVERR  001174RG    003 S.FDB = 000140          TRNS6   002444R     003 T.NAM   000006          .CLOSE= ****** G
RCVSE   006732RG    003 S.FNAM= 000006          TRNS7   002470R     003 T.OFF   000060          .CSI1 = ****** G
RCVSWT= 000014 G        S.FNB = 000036          TROKT   002652R     003 T.PCB   000046          .CSI2 = ****** G
RCVSW1= 000014 G        S.FNBW= 000017          TRPINS  001442R     003 T.PRI   000002          .FIND = ****** G
```

```
RCSRSXM.F001     MACRO M1104  06-DEC-77 08:21  PAGE 52-3
SYMBOL TABLE

.FSRCB= ****** G          ...PC1= 000336R      003 ...PC2= 000532R      003 ...PC3= 000336R      003 ...TPC= 000020
.PARSE= ****** G

.  ABS.   000070      000
           000002      001
ISR        000000      002
MAIN       007130      003
$$FSR1     002040      004
MESAG2     000256      005
ERRORS DETECTED:   0

VIRTUAL MEMORY USED:   7589 WORDS  ( 30 PAGES)
DYNAMIC MEMORY:   8090 WORDS  ( 31 PAGES)
ELAPSED TIME:   00:01:53
 256,020 GET;27, 256,030 GET/CRF/-SP=RSXMC,EXEMC/ML,UAPRXM,CTXMAC,GET
```

GET CREATED BY MACRO ON 6-DEC-77 AT 08:22 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL		VALUE		REFERENCES							
ACK	=	000040	G	5-143	#5-148						
ACKPEN		001470	RG	21-389	#21-420						
ASROUT	=	******	G	5-140	37-1005						
ASSGNX		001505	RG	21-391	#21-448						
A$$CHK	=	000000		#1-35							
A$$CPS	=	000000		#1-29	#1-47						
A$$NSI	=	000000		#1-48							
A$$PRI	=	000000		#1-57							
A$$TRP	=	000000		#1-49	1-66						
BADCS		002400	R	28-762	#28-771						
BASELN	=	000000	G	13-250	#13-253	50-1304	50-1310	50-1315	50-1322	50-1337	50-1343	50-1349
BDFAIL		002616	R	#33-831	33-839	33-841						
BINEOT	=	177467	G	13-260	#13-269						
BINESC	=	000047	G	13-261	#13-266						
BINITB	=	000037	G	13-260	#13-268						
BINTTD	=	026402	G	13-261	#13-265						
BIN4	=	000364	G	13-261	#13-267						
BISIN		002572	R	#33-825	37-1009						
BISIN2		003256	R	#37-1003	37-1006						
BISPAR		004152	RG	50-1297	#50-1322						
BPTINS		001362	R	12-238	#14-342						
BUFOVF		002212	R	#25-728	36-968						
CHANL		003752	RG	#44-1144							
CHANL2		003760	RG	#44-1147							
CHARL		003770	RG	#44-1151							
CLOSER	=	******	G	5-138	42-1093						
CLOSET	=	******	G	5-138	33-834	42-1092					
CLRALL		003636	R	37-1000	42-1091	#42-1109					
CLRMST		003642	RG	42-1052	#42-1110						
CLRRSW		003710	RG	43-1126	#43-1127						
CMBFAV		001502	RG	21-391	#21-440	37-1003					
CMBFPN		001503	RG	21-391	#21-442	*42-1112					
CMBLST		001526	RG	22-480	#22-481						
CMBMAX	=	000016	G	13-250	#13-256						
CMBUFE	=	006066	RG	51-1364	#51-1375	51-1379					
CMBUF1		004214	RG	21-471	22-481	51-1364	#51-1367	51-1369	51-1375		
CMBUF2	=	004422	R	22-482	#51-1369	51-1370					
CMBUF3	=	004630	R	22-483	#51-1370	51-1371					
CMBUF4	=	005036	R	22-484	#51-1371	51-1372					
CMBUF5	=	005244	R	22-485	#51-1372	51-1373					
CMBUF6	=	005452	R	22-486	#51-1373	51-1374					
CMBUF7	=	005660	R	22-487	#51-1374						
CMCLST		001562	RG	22-480	#22-498						
CMFLST		001600	RG	22-480	#22-506						
CMPAR		004200	RG	*33-843	50-1297	#50-1343					
CMPLST		001544	RG	22-480	#22-490						
COCOM		003342	R	38-1017	#38-1021						
COCOM1		003632	R	38-1022	38-1045	42-1065	#42-1105				
COCOM2		003434	R	38-1035	#38-1044						
CODE		001461	RG	21-388	#21-398	33-848	37-983				
COLAD		003766	RG	#44-1150							
COLP1		003754	RG	#44-1145							

SYMBOL		VALUE		REFERENCES							
COLP2		003762	RG	#44-1148							
COLTT	=	******	G	8-186							
COMATB		004030	R	36-956	42-1090	#48-1240					
COMAV	=	000007	G	13-250	#13-255	13-256	13-256	21-440			
COMCPL		002316	R	23-631	#28-747						
COMDD		003076	R	34-865	34-867	#36-947					
COMERR		002276	RG	25-726	#25-736	28-772	30-785	36-961	42-1075		
CR	=	000015		#13-276							
CRBASX		001524	RG	21-393	#21-471						
CRBFAV		001501	RG	21-391	#21-436						
CROUT		001622	RG	23-518	#23-540	25-740	30-790	30-796	33-835	34-896	
CROUTX		001616	RG	#23-539	28-760	34-893	42-1105				
CSIERR		002406	RG	28-770	28-771	#28-774					
CS.INP	=	000001		34-862							
CS.OUT	=	000002		28-763							
CTEMT	=	******	GX	14-332							
CTIO		002330	R	28-759	#28-761						
C$$CKP	=	000000		#1-32	1-65	#1-95					
C$$ORE	=	004000		#1-96							
C$$RSH	=	177564		#1-87							
C.BODE	=	******	GX	23-567							
C.CMLD	=	000002		*28-761	*28-761	36-953					
C.DEVD	=	000006		28-764	34-863						
C.FILD	=	000016		28-774	28-775						
C.SGET	=	******	GX	14-324							
C.SIZE	=	000054		28-761	28-763	34-862	51-1492				
C.SPS	=	******	G	8-190	23-518						
C.SPUT	=	******	GX	14-318							
C.SREG	=	******	GX	14-317	14-323	14-327	14-331				
C.SRES	=	******	GX	14-320	14-334						
C.STAB	=	******	GX	23-566							
C.STIN	=	******	GX	14-328							
DEVRT		001456	RG	20-380	#20-381	28-764					
DFBLK		000000	RG	#12-198	12-202	12-207					
DFNLP		000476	RG	12-222	#12-223						
DIE		002206	RG	14-340	14-341	14-342	14-343	14-344	14-345	14-346	23-635
				23-642	23-685	#23-686	25-735	37-991			
DLEEOT		004164	R	50-1312	#50-1331						
DLEPAR		004136	R	42-1077	#50-1310						
DONE	=	******	G	5-142	23-686						
DSCHAN		003502	R	#42-1072	48-1253						
DSCOUT		003520	R	42-1074	#42-1076						
DSW		000000	R	#5-108							
D$$BUG	=	177514		#1-38							
D$$H11	=	000002		#1-126							
D$$IAG	=	000000		#1-50							
D$$ISK	=	000000		#1-33							
D$$L11	=	000001		#1-125							
D$$M11	=	000001		#1-127							
D$$SHF	=	000000		#1-39							
D$$YNC	=	000000		#1-34							
D$$YNM	=	000000		#1-38							

SYMBOL		VALUE		REFERENCES							
EBCHAN		003312	R	37-979	#38-1014	48-1247					
ELATCH		001471	RG	21-389	#21-422	*33-824					
EMTERR		001320	R	14-319	14-325	14-329	#14-334				
EMTGD		001246	R	#14-320	14-326	14-330	14-333				
EMTGET		001256	RG	14-315	#14-323	23-581					
EMTINT		001272	RG	14-315	#14-327						
EMTPUT		001234	RG	14-315	#14-317	30-791	33-844	33-849	37-984	42-1077	42-1089
EMTTIM		001306	RG	14-315	#14-331						
ENBPAR		004130	RG	*33-846	*33-847	*33-848	33-849	*37-982	*37-983	37-984	50-1297	#50-1304
ENDPEN		001472	RG	21-389	#21-423	*42-1113					
ENQM		004124	R	#50-1299	50-1317						
ENQPAR		004144	R	33-844	#50-1315						
ENTRY		001764	RG	23-518	*23-579	23-580	23-583	#23-589			
EOTFLG		001476	RG	21-390	#21-430	*42-1078	*42-1115				
EOTPEN		001474	RG	21-390	#21-426	*42-1114					
EOTSEQ		004160	RG	*37-1002	50-1297	#50-1328					
ERRHAN	=	******	G	5-141	48-1265						
ESC	=	000033	G	13-260	#13-263						
ETXFLG		001475	RG	21-390	#21-428						
EXBHAN		003332	R	#38-1019	48-1245						
EXPREC		006276	R	12-200	12-204	12-219	*51-1385				
E$$DVC	=	000000		#1-90	1-203	1-338					
E$$NSI		000000		#1-91							
E$$PER	=	000000		#1-92							
E$$XPR	=	000000		#1-58							
FDBIN		000036	RG	#12-199	25-731	25-731	*38-1032	*38-1033	*38-1040	*38-1041	
FDBOU0		000176	RG	#12-203	25-732	25-732	44-1144				
FDBOU1		000336	RG	#12-208	25-733	25-733	44-1147				
FD.CR	=	******	GX	12-206	12-221	43-1134					
FD.PLC	=	******	GX	12-200	12-204	12-219					
FILDES		006734	RG	28-761	28-761	28-774	28-775	36-953	51-1365	#51-1492	
FLAGS		001462	RG	21-388	#21-400	*33-842	33-847	37-982	*38-1015	*38-1016	*38-1020	*42-1116
FSTFIL	=	******	G	5-139	33-829						
FSTGET	=	******	G	5-139	33-840						
F$$LPP	=	******		1-373							
F$$LVL	=	000001		#1-31							
F.DFNB	=	000046		12-202	12-207	12-222					
F.LUN	=	000042		12-202	12-207	12-222					
F.MBCT	=	000054		12-201	12-205	12-220	*38-1033	*38-1041			
F.RACC	=	000016		12-200	12-204	12-219					
F.RATT	=	000001		12-206	12-221						
F.RSIZ	=	000002		12-206	12-221	*38-1032	*38-1040				
F.RTYP	=	000000		12-206	12-221						
F.URBD	=	000020		12-200	12-200	12-204	12-204	12-219	12-219		
GETENT		001622	RG	23-518	#23-542	23-634	23-641	25-729	33-852	36-972	
GETEXP		001774	R	23-582	#23-600						
GETO1		001654	R	23-544	#23-547						
GNEMDE		003350	R	#38-1028	48-1256						
GOABRT	=	******	G	5-140	42-1066						
GOGETE		003514	R	38-1014	38-1019	38-1028	38-1037	42-1059	#42-1075		
GOTG		002764	R	33-827	#33-852						
GOTGET		001616	RG	#23-538							

```
SYMBOL  VALUE          REFERENCES
G$$TPP  = 000000       #1-54
G$$TTK  = 000000       #1-55
HFCACT    001466 RG    21-389    #21-416
HFCREC    006066 RG    51-1364   #51-1380
IE.EOF  = ******  GX    23-633
ILLINS    001426 R     12-241    #14-345
INCRCV    001477 RG    21-393    #21-432
INITAL  = ******  G     5-138     14-308
INITF     001467 RG    21-389    #21-418   *33-851
INPCNT    001514 RG    21-392    #21-467   *42-1110
INPCPL  = ******  G     5-141     36-970
IOERR     001510 RG    21-392    #21-453    42-1062  *42-1064  *42-1088
IOST      000674 R     12-225    #12-229    23-628    23-630    23-633    28-758
IOTINS    001376 R     12-239    #14-343
IO.KIL  = ******  GX    12-226
IO.RLB  = ******  GX    12-225
IO.WLB  = ******  GX    12-224     12-227
ITBM      004126 RG    50-1297   #50-1301
I$$RAR  = 000000       #1-61
I$$RDN  = 000000       #1-36
JMPTBL    003136 R     23-586    #36-967
JOINT     002722 R     #33-846    37-1010
JOINTI    003272 R     37-1004   #37-1
JOINTX    003304 R     37-1008   #37-1010
KBIBUF    007010 RG    12-225     51-1365  #51-1494
K$$CNT  = 172542       #1-102
K$$CSR  = 172540       #1-101
K$$LDC  = 000024       #1-104
K$$TPS  = 000764       #1-103
LD$DB   = 000000       #1-106
LD$DK   = 000000       #1-109
LD$DS   = 000000       #1-112
LD$DT   = 000000       #1-114
LD$LP   = 000000       #1-116
LD$MM   = 000000       #1-119
LD$PP   = 000000       #1-121
LD$PR   = 000000       #1-122
LF      = 000012       #13-277
LININT    001206 RG    14-285    #14-300   52-1506
LPSIZE  = 000206  G    13-250    #13-251   50-1325  51-1369  51-1370  51-1371  51-1372  51-1373  51-1374
                       51-1375    51-1380  51-1385
LUKUP   = ******  G     5-138     30-813
L$$ASG  = 000000       #1-62
L$$DRV  = 000000       #1-51
L$$F11  = 000001       #1-118
L$$SI1  = ******        1-311
L$$11R  = 000000       #1-117
MCB       001463 RG    21-388    #21-407   *38-1029  *38-1038
MDMPAR    004172 R     30-791     30-792   #50-1337
MDTRY     002474 R     30-788    #30-791
MEMPRO    001346 R     12-237    #14-341
MODE      001465 RG    21-388    #21-412    33-826    34-870    36-962    42-1060  *42-1109
```

SYMBOL	VALUE		REFERENCES							
MSGDON	= ******	G	5-140	36-971						
M$$CRB	= 000124		#1-135							
M$$CRX	= 000000		#1-131							
M$$EXT	= 000000		#1-99							
M$$FCS	= 000000		#1-30							
M$$MGE	= 000000		#1-37							
M$$MUP	= 000000		#1-43							
M$$OVR	= 000000		#1-130							
NAK	= 000100	G	5-143	#5-149						
NAKCNT	001516	RG	21-393	#21-468	*42-1111					
NRCVSF	= 000000		#2-5	5-116	51-1422					
NTRNSF	= 000002		#2-4	5-112	51-1401					
N$$LDV	= 000001		#1-129							
N$$MOV	= 000021		#1-60							
N.DVNM	= 000032		12-198	12-223						
N.FNAM	= 000006		12-198	12-223						
N.FTYP	= 000014		12-198	12-223						
N.UNIT	= 000034		12-198	12-223						
ODDADD	001332	R	12-236	#14-340						
OFFHAN	003546	RG	42-1052	#42-1085	48-1241					
OFHAN1	003576	R	14-309	#42-1090						
ONLHAN	003152	RG	37-978	#37-979	48-1243					
ONLINE	001460	RG	21-388	#21-396	23-671	30-783	*37-980	42-1058	42-1073	*42-1086
PAR$$$	= 000027		#25-731	25-731	#25-732	25-732	#25-733	25-733	#28-761	28-761
PENTER	= ******	GX	34-891							
POINT	= ******	G	5-139	33-838						
POINT1	000700	RG	#12-230							
POINT2	000702	RG	#12-231							
POINT3	000704	RG	#12-232							
PREND	= 000255	R	#52-1501	52-1502						
PRGERR	002232	R	#25-730	33-853	36-967					
PRINTX	001507	RG	21-392	#21-450						
PRMTA	000247	R	12-224	#52-1500	52-1502					
PRMTL	= 000005		23-670	#52-1502						
PROMPT	002142	RG	23-539	23-545	#23-669	25-728	25-739	30-789	30-794	33-821 33-831
			34-895	37-997						
PS	= 177776		#5-125							
PUTERR	002770	R	33-845	33-850	#33-853					
P$$D70	= 000000		#1-100	#1-105						
P$$GMX	= 000000		#1-42							
P$$LAS	= 000000		#1-40							
P$$P11	= 000001		#1-123							
P$$P45	= 000000		#1-97							
P$$RFL	= 000000		#1-53							
P$$RTY	= 000000		#1-93							
P$$R11	= 000001		#1-124							
P$$SRF	= 000000		#1-41							
QBLKM	004114	RG	*38-1044	49-1280	#49-1286					
QCODE	004120	R	*38-1021	#49-1288						
QCONT	003310	R	#37-1011	42-1117						
QDPBK	000614	R	#12-226	23-652						

SYMBOL		VALUE		REFERENCES							
QDPBM		000644	RG	#12-227	23-660	*28-774	*28-775	28-776			
QDPBR		000564	R	#12-225	23-676	28-761					
QDPBW		000534	R	#12-224	23-669						
QEND		004124	RG	49-1280	#49-1290	49-1291	49-1292				
QLENTH	=	000020	G	49-1280	#49-1292						
QLNGTH	=	000030	G	49-1280	#49-1291						
QMODE		004110	RG	*42-1117	49-1280	#49-1284					
QONL		004034	R	37-981	#48-1242						
QSTAT		004104	RG	*37-981	*42-1090	49-1280	#49-1282	49-1291	49-1292		
QUEHAN	=	******	G	5-141	48-1263						
Q$$OPT	=	000017		#1-59							
Q.IOFN	=	000002		12-224	12-225	12-226	12-227				
Q.IOPL	=	000014		*23-661	*23-662	*23-670	*23-673	28-761	*28-774	*28-775	
RADCX	=	******	GX	36-954							
RATT		004006	RG	#45-1198							
RATT1		004024	RG	*43-1134	#45-1211						
RCHN		004002	RG	*42-1085	#45-1196						
RCHN1		004020	RG	#45-1209							
RCVBLK		006656	RG	34-869	42-1095	51-1365	#51-1407	51-1407			
RCVCNT		001512	RG	21-394	#21-456						
RCVERR		001174	RG	14-285	#14-293						
RCVSE		006732	RG	#51-1426							
RCVSWT	=	000014	G	#45-1200	51-1421						
RCVSW1	=	000014	G	43-1127	#45-1213						
RCVTIM	=	000177	G	13-250	#13-257						
RCVTMT		001200	RG	14-286	#14-295						
RECSIZ		001520	RG	21-393	#21-469	*38-1030	*38-1043				
RECS1		003756	RG	#44-1146							
RECS2		003764	RG	#44-1149							
RESP		004153	RG	50-1297	#50-1323						
RPEND		003056	R	34-871	#34-895						
RPEND1		003072	R	34-892	#34-896						
RSVDIN		001412	R	12-240	#14-344						
RSX	=	000001		#2-2	5-27	5-99	5-122	5-150	6-159	8-187	12-196	14-288
				14-302	14-339	20-379	23-537	23-612	25-730	28-757	28-769	33-821
				34-861	34-890	36-952	37-988	37-996	38-1031	38-1039	42-1094	42-1104
				43-1120	44-1143	45-1188	51-1384	51-1399	51-1408	51-1420	51-1491	52-1498
RSX11M	=	000001		#2-3	5-28	5-35	5-100	23-548	23-613	23-622	23-625	23-653
RTMDE		003404	R	#38-1037	48-1258							
RTYP		004004	RG	#45-1197								
RTYP1		004022	RG	*43-1133	#45-1210							
RT11	=	******		5-32	5-96	5-144	6-171	8-181	20-361	21-461	21-472	23-520
				23-575	23-600	23-689	25-736	28-747	30-798	33-828	34-857	34-873
				35-898	36-947	37-985	37-992	42-1053	42-1097	42-1101	44-1155	45-1216
				48-1266	50-1354	51-1390	51-1413	51-1429				
RVAL		004010	RG	#45-1199								
RVAL1		004026	RG	*43-1132	#45-1212							
RVIPEN		001500	RG	21-390	#21-435							
R$$DER	=	000000		#1-86								
R$$EIS	=	******		1-358								
R$$EXV	=	000000		#1-79								
R$$JPO	=	000000		#1-107								

SYMBOL	VALUE		REFERENCES					
R$$JP1	= 000001		#1-108					
R$$JS1	= 000001		#1-113					
R$$KWC	= 000000		#1-110					
R$$K11	= 000001		#1-111					
R$$NDC	= 000003		#1-80					
R$$NDH	= 000226		#1-81					
R$$NDL	= 000001		#1-82					
R$$SND	= 000000		#1-56	#1-94				
R$$11M	= 000000		#1-136					
R.VAR	= ******		12-206	12-221	43-1133			
SIBEFN	= 000040		#5-156	23-563	23-565	23-570	23-639	
SLEEP	002060	R	23-621	#23-638				
SNDEOT	003476	RG	42-1052	42-1061	42-1063	#42-1066		
SSTVEC	000706	R	#12-234	14-303				
STEQM	004162	R	*37-1001	#50-1329				
STKLMT	000732	R	#14-289	23-543	*23-546			
STKSET	001174	RG	14-285	#14-292	14-301			
STOREX	001506	RG	21-392	#21-449				
SWRBN	003776	RG	#45-1194	45-1200				
SWRB1	004014	RG	43-1128	#45-1207	45-1213			
SWRDF	004012	RG	#45-1201					
SWREN	003774	RG	#45-1192					
SWRSL	004000	RG	#45-1195					
SWRSUP	001522	RG	21-393	#21-470				
SWRS1	004016	RG	#45-1208					
SWTBN	003772	RG	#45-1191	45-1193				
SYSERR	002076	R	#23-642	23-675	23-677			
S$$WPC	= 000036		#1-83					
S$$WPR	= 000005		#1-84					
S$$WRG	= 000000		#1-89					
S$$YDF	= ******		1-1	1-438				
S$$YSZ	= 024000		#1-98					
S.BFHD	= 000020		12-228	12-228				
S.FDB	= 000140		12-199	12-199	12-203	12-203	12-208	12-208
S.FNAM	= 000006		12-198	12-198	12-223	12-223		
S.FNB	= 000036		12-198	12-223	51-1400	51-1404	51-1404	51-1421
S.FTYP	= 000002		12-198	12-198	12-223	12-223		
S.VTTA	= 000002		12-234					
TABSST	000714	RG	12-234	#12-236				
TAGE1	000010	R	14-340	#14-340				
TAGE10	000103	R	23-545	#23-545				
TAGE11	000111	R	23-635	#23-635				
TAGE12	000136	R	23-642	#23-642				
TAGE13	000146	R	25-728	#25-728				
TAGE14	000156	R	25-735	#25-735				
TAGE15	000165	R	25-739	#25-739				
TAGE16	000176	R	30-789	#30-789				
TAGE17	000211	R	30-794	#30-794				
TAGE2	000020	R	14-341	#14-341				
TAGE20	000226	R	33-831	#33-831				
TAGE21	000237	R	34-895	#34-895				
TAGE22	000247	R	37-991	#37-991				

SYMBOL	VALUE		REFERENCES								
TAGE3	000027	R	14-342	#14-342							
TAGE4	000036	R	14-343	#14-343							
TAGE5	000046	R	14-344	#14-344							
TAGE6	000055	R	14-345	#14-345							
TAGE7	000065	R	14-346	#14-346							
TAGNUM	= 000022		#2-8	14-340	#14-340	14-340	14-341	#14-341	14-341	14-342	#14-342
			14-342	14-343	#14-343	14-343	14-344	#14-344	14-344	14-345	#14-345
			14-345	14-346	#14-346	14-346	23-545	#23-545	23-545	23-635	#23-635
			23-635	23-642	#23-642	23-642	25-728	#25-728	25-728	25-735	#25-735
			25-735	25-739	#25-739	25-739	30-789	#30-789	30-789	30-794	#30-794
			30-794	33-831	#33-831	33-831	34-895	#34-895	34-895	37-991	#37-991
			37-991								
TAG1	000000	R	14-340	14-340	#14-340						
TAG10	000065	R	23-545	23-545	#23-545						
TAG1	000103	R	23-635	23-635	#23-635						
TAG1.	000111	R	23-642	23-642	#23-642						
TAG13	000136	R	25-728	25-728	#25-728						
TAG14	000146	R	25-735	25-735	#25-735						
TAG15	000156	R	25-739	25-739	#25-739						
TAG16	000165	R	30-789	30-789	#30-789						
TAG17	000176	R	30-794	30-794	#30-794						
TAG2	000010	R	14-341	14-341	#14-341						
TAG20	000211	R	33-831	33-831	#33-831						
TAG21	000226	R	34-895	34-895	#34-895						
TAG22	000237	R	37-991	37-991	#37-991						
TAG3	000020	R	14-342	14-342	#14-342						
TAG4	000027	R	14-343	14-343	#14-343						
TAG5	000036	R	14-344	14-344	#14-344						
TAG6	000046	R	14-345	14-345	#14-345						
TAG7	000055	R	14-346	14-346	#14-346						
TASKAD	= ******	GX	23-562	23-571	23-617						
TDONE	001473	RG	21-390	#21-425	*33-822						
TERHAN	003442	RG	42-1052	#42-1058	48-1251						
TERPAR	004206	R	42-1089	#50-1349							
TERR	002530	R	#30-795								
TER1	= ******	G	5-142	42-1079							
TRNSBL	006504	RG	30-786	30-795	33-833	42-1100	51-1364	#51-1387	51-1387		
TRNSE	006654	RG	#51-1405								
TRNS4	002432	R	28-766	#30-783							
TRNS5	002540	R	30-793	#30-797							
TRNS6	002444	R	30-784	#30-786							
TRNS7	002470	R	#30-790	30-814							
TROKT	002652	R	33-830	#33-836							
TRPINS	001442	R	12-242	#14-346							
TRRTNF	001504	RG	#21-444	23-518	*33-823	*42-1072	*42-1076				
TRYREC	002774	R	28-767	#34-857							
TTERFN	= 000020		#5-155								
TTKLFN	= 000004		#5-153								
TTYDFN	= 000002		#5-152								
TTYDFN	= 000010		#5-154	23-620	23-626	23-639					
TURNAR	003236	RG	37-978	#37-998							
TURNON	003232	R	37-989	#37-997							

```
SYMBOL  VALUE          REFERENCES
TURN1     003236 RG     37-978    #37-1000
TVAB      001464 RG     21-388    #21-411
TXPEN     001511 RG     21-392    #21-454    *33-825    *33-832    33-836    37-1007   *42-1087
TYPMSG    002112 RG     14-340     14-341     14-342     14-343    14-344    14-345     14-346     23-545     23-635
                        23-642    #23-652     25-728     25-735    30-794    33-831     37-991
TYPMS1    002120 RG    #23-660     25-739     30-789     34-895
T$$ACR  = 000000       #1-63
T$$BTW  = 000000       #1-45      #1-64
T$$BUF  = 000000       #1-65
T$$CCA  = 000000       #1-66
T$$CCO  = 000000       #1-67
T$$CTR  = 000000       #1-68
T$$C11  = 000001       #1-115
T$$GMC  = 000000       #1-69
T$$GTS  = 000000       #1-70
T$$J16  = 000001       #1-120
T$$KMG  = 000000       #1-52
T$$LWC  = 000000       #1-71
T$$MAN  = 006307       #1-128
T$$RNF  = 000000       #1-46      #1-72
T$$RPR  = 000000       #1-73
T$$RST  = 000000       #1-44      #1-74
T$$RUB  = 000000       #1-75
T$$SYN  = 000000       #1-76
T$$TRW  = 000000       #1-77
T$$VBF  = 000000       #1-78
T.EFLG    000022        23-563    *23-565    *23-570    *23-572    23-618    *23-626
UBFCNT    001204 RG     14-285    #14-297
UIBEFN  = 000001       #5-151     23-572     23-639
USRTIM    001622 RG    #23-541     36-969
U$$MHI  = 000040       #1-133
U$$MLO  = 120000       #1-134
U$$MRN  = 170324       #1-132
V$$CTR  = 000400       #1-137
XCEP    = 077405       #13-270     50-1299
XDEOT   = 002020       #13-272     50-1331
XEOTS   = 077404  G     13-261    #13-271     37-1002    50-1328
XFEREN    004100 RG     48-1238   #48-1270
XITB    = 000037  G     13-260    #13-274     50-1301
XMTERR    001176 RG     14-285    #14-294
XMTSWT  = 000004  G    #45-1193    51-1400    51-1404    51-1404
XMTTMT    001202 RG     14-286    #14-296
XPAD    = 000177       #13-273     50-1332
XSTEQ   = 002402  G     13-260    #13-275     37-1001    50-1329
X$$DBT  = 000000       #1-85
$$$ARG  = 000003       #12-224     12-224    #12-224     12-224    #12-224     12-224    #12-224     12-224     12-224
                       #12-225     12-225    #12-225     12-225    #12-225     12-225    #12-225     12-225     12-225
                       #12-226     12-226     12-226    #12-227     12-227    #12-227     12-227    #12-227     12-227
                       #12-227     12-227     12-227
$$$GLB  = ******        12-224     12-225     12-226     12-227     12-234
.CLOSE  = ******  G     25-731     25-731     25-732     25-732     25-733     25-733
.CSI1   = ******  G     28-761     28-761
```

SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE	REFERENCES								
.CS12	= ****** G	28-763	28-763	34-862	34-862					
.FIND	= ****** G	8-188								
.FSRCB	= ****** G	12-228								
.PARSE	= ****** G	8-189								
...PC1	= 000336 R	12-198	12-198	#12-198	12-198	12-198	12-198	12-198	12-198	#12-198
		#12-199	12-200	12-200	12-200	12-201	12-202	12-202	#12-203	12-204
		12-204	12-204	12-205	12-206	12-206	12-206	12-207	12-207	#12-208
		12-219	12-219	12-219	12-220	12-221	12-221	12-221	12-222	12-222
		12-223	12-223	#12-223	12-223	12-223	12-223	12-223	12-223	#12-223
...PC2	= 000532 R	#12-198	12-198	12-198	#12-198	12-198	12-198	#12-198	12-198	#12-200
		12-200	#12-200	12-200	#12-200	12-200	#12-201	12-201	#12-202	12-202
		#12-202	12-202	#12-204	12-204	#12-204	12-204	#12-204	12-204	#12-205
		12-205	#12-206	12-206	#12-206	12-206	#12-206	12-206	#12-207	12-207
		#12-207	12-207	#12-219	12-219	#12-219	12-219	#12-219	12-219	#12-220
		12-220	#12-221	12-221	#12-221	12-221	#12-221	12-221	#12-222	12-222
		#12-222	12-222	#12-223	12-223	12-223	#12-223	12-223	12-223	#12-223
		12-223								
..PC3	= 000336 R	#12-198	12-198	#12-223	12-223					

MACRO NAME	REFERENCES									
ASRBUF	#4-28									
BDOFF$	#12-228									
CALL	#1-144	#6-168	23-539	33-821	37-997					
CALLR	#1-170									
CBYTE$	#12-200	12-200	#12-201	12-201	12-201	12-201	#12-202	12-202	12-202	#12-204
	12-204	#12-205	12-205	12-205	12-205	#12-206	12-206	12-206	#12-207	12-207
	12-207	#12-219	12-219	#12-220	12-220	12-220	12-220	#12-221	12-221	12-221
	#12-222	12-222	12-222							
CLOSE$	#6-163	#25-731	#25-732	#25-733						
CRASH	#1-178									
CSI$	#6-161	#6-162	#28-761	#28-763	#34-862					
CSI$1	#6-166	#28-761								
CSI$2	#6-166	28-763	34-862							
CWOPD$	#12-198	12-198	12-198	#12-200	12-200	12-200	#12-201	12-201	#12-202	12-202
	12-202	12-202	#12-204	12-204	12-204	#12-205	12-205	#12-206	12-206	12-206
	12-206	#12-207	12-207	12-207	12-207	#12-219	12-219	12-219	#12-220	12-220
	#12-221	12-221	12-221	12-221	#12-222	12-222	12-222	12-222	#12-223	12-223
	12-223									
DEF$L	#12-228									
DIRSV$	#1-186									
DIR$	#6-167	#14-303	#23-639	#23-639	#23-652	#23-663	#23-674	#23-676	#28-776	
DRSTS	#1-194									
ERROR	#2-9	#23-545	#25-728	#30-794	#33-831					
ERRORK	#2-16	14-340	14-341	14-342	14-343	14-344	14-345	14-346	23-545	23-635
	23-642	25-728	25-735	30-794	33-831	37-991				
ERRORM	#2-32	25-739	30-789	34-895						
ERRORP	#2-47	#25-739	#30-789	#34-895						
ERR$	#25-731	#25-731	#25-732	#25-732	#25-733	#25-733				
FATAL	#2-53	#14-340	#14-341	#14-342	#14-343	#14-344	#14-345	#14-346	#23-635	#23-642
	#25-735	#37-991								
FDAT$A	#6-164	#12-206	#12-221							
FDAT$R	#6-165									
FDBDF$	#6-164	12-199	12-203	12-208						
FDBF$A	#6-164	12-201	12-205	12-220						
FDBK$A	#6-164									
FDBSZ$	#12-199	12-199	#12-203	12-203	#12-208	12-208				
FDOFF$	#12-200	12-200	#12-201	12-201	#12-202	12-202	#12-204	12-204	#12-205	12-205
	#12-206	12-206	#12-207	12-207	#12-219	12-219	#12-220	12-220	#12-221	12-221
	#12-222	12-222								
FDOP$A	#6-164	#12-202	#12-207	#12-222						
FDRC$A	#6-164	#12-200	#12-204	#12-219						
FSRSZ$	#6-164	12-228								
GETS	#4-43	23-581								
GTUCB$	#1-290									
INTLB	#1-204									
INTSE$	#1-214									
INTSV$	#1-256									
LDFDB$	#25-731	25-731	#25-732	25-732	#25-733	25-733				
LDRO$	#28-761	#28-761	#28-763	#28-763	#34-862	#34-862				
LINIT	#4-23									
MFPS	#1-312									
MOV$	#23-639	23-639								

MACRO NAME REFERENCES

MTPS #1-315
NBOFF$ #12-198 12-198 #12-223 12-223
NMBLK$ #6-165 12-198 12-223
OFF$ #12-224 #12-225 #12-226 #12-227 #12-234
PUTD #4-52 #33-844 #42-1077
PUTMC #4-38 30-791
PUTTC #4-33 33-849 37-984 42-1089
PUTTM #4-47
QDPB$ #12-224 12-224 #12-225 12-225 #12-226 12-226 #12-227 12-227
QIO$ #6-167 12-224 12-225 12-226 12-227
RAD50$ #12-198 12-198 12-198 #12-223 12-223 12-223
RETURN #1-349
SAVNR #1-329
SCBLB #1-339
SOB #1-359
SVTK$ #6-169 12-234
SWSTK$ #1-369
TCBDF$ #5-104 5-105
WTLOS$ #6-163 #23-639
WTSE$S #6-163

.MAIN. MACRO M1104 06-DEC-77 08:24
TABLE OF CONTENTS

2- 1 ASSEMBLE FOR USE WITH AN UNMAPPED SYSTEM

INTHND MACRO M1104 06-DEC-77 08:24 PAGE 1

```
                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM


                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                                        .SBTTL  ASSEMBLE FOR USE WITH AN UNMAPPED SYSTEM
                                ;
```

```
INTHND  MACRO M1104  06-DEC-77 08:24  PAGE 3
ASSEMBLE FOR USE WITH AN UNMAPPED SYSTEM

                                        .TITLE  INTHND
                                        .IF DF  M$$MGE
                                        .IDENT /0003M/
                                        .IFF
                                        .IDENT  /0003U/
                                        .ENDC
                                ;
                                ; COPYRIGHT (C)  1975, 1976 ,1977
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                                ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                                ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                                ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                                ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                                ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                                ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                                ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                                ;
                                ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                                ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                                ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                                ;
                                ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                                ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                                ; DEC.
                                ;
                                ; VERSION 0
                                ;
                                ; ROBERT H.  ROSENBAUM  30-JULY-74
                                ;
                                ; LAST MODIFICATION ON 24-JUNE-75
                                ;
                                ; VERSION 1
                                ;
                                ; CLARISE E.  PATTON  SEPT 1977
                                ;
                                ; MODIFIED TO ACCEPT EITHER APR4 OR APR5
                                ;
                                ;
                                ; INTERRUPT RELATED ROUTINES FOR RSX-11M
                                ;
                                ; REGISTER DEFINITIONS
                                ;
  177776                                PS=177776
                                        .IF DF M$$MGE
                                        UISAR4=177650
                                        KISAR4=172350
                                        UISAR5=177652
                                        KISAR5=172352
                                        .ENDC
                                ;
                                ; MACRO CALLS
                                        .MCALL  CALL
                                ;
000000                                  .PSECT  MAIN
                                ;
```

```
                                        ;+
                                        ; **-..CINT-CONNECT TO DEVICE VECTOR
                                        ;
                                        ; THIS ROUTINE IS CALLED TO ENABLE A USER TASK TO HANDLE INTERRUPTS
                                        ; FROM A PARTICULAR DEVICE (SIMULATES THE RSX-11D EXEC ROUTINE
                                        ; BY THE SAME NAME)
                                        ;
                                        ; INPUTS:
                                        ;
                                        ;       R0 -- INTERRUPT TRAP ADDRESS
                                        ;       R1 -- ENTRY POINT OF INTERRUPT SERVICE ROUTINE
                                        ;             (ADDRESS MUST BE WITHIN USER APR'S RANGE)
                                        ;       R3 -- BITS 5 - 7 PRESCRIBE THE PRIORITY
                                        ;
                                        ; CALLING SEQUENCE:
                                        ;
                                        ;       JSR     PC,..CINT
                                        ;
                                        ; OUTPUTS:
                                        ;
                                        ;       THIS ROUTINE WILL CHECK TO SEE IF IT IS TO USE APR4 (APR5 IS
                                        ;       DEFAULT), THEN WILL REQUEST A BUFFER FROM THE EXEC, STORE THE BUFFER
                                        ;       ADDR IN THE VECTOR ADDR SUPPLIED BY THE CALLER, MOVE THE INSTRUCTIONS
                                        ;       STARTING AT "INTAD" THROUGH "INTEND" INTO THE BUFFER FILLING IN
                                        ;       THE ADDR CONTAINED IN USER APR4 AND THE USER'S ISR ADDR
                                        ;       AND FINALLY RETURN TO THE USER WITH ALL REGISTERS UNCHANGED.
                                        ;       THE "C" BIT WILL BE SET IF:
                                        ;       1) THE VECTOR IS IN USE (I.E. THE ADDR IN THE VECTOR IS NOT
                                        ;          THAT OF THE NONSENSE INTERRUPT ROUTINE ($NONSI) OR
                                        ;       2) NO BUFFERS ARE AVAILABLE
                                        ;
                                        ;-
                                        ;
000000  004567  000000G        ..CINT: JSR     R5,C.SREG       ;SAVE THE REGISTERS
                                        .IF DF M$$MGE           ;IF MEMORY MANGEMENT
                                        CMP     #STKSET,#120000         ;WHICH REGISTER USED?
                                        BGT     5$                      ;REGISTER 5
                                        MOV     #UISAR4,UISAR   ;MOVE IN REGISTER 4
                                        MOV     #KISAR4,KISAR           ;
                                5$:
                                        MOV     @UISAR,INTAD+1.KA       ;PUTS CONTENTS OF PAR INTO
                                                                        ;MOVE INSTRUCTION
                                        .ENDC
000004  010167  000162                  MOV     R1,INTAD:INTJSR ;STORE ADDR INTO JSR INTS
000010  005103                          COM     R3              ;COMPLEMENT THE PRIORITY
000012  042703  177437                  BIC     #177437,R3      ;CLEAR ALL BUT PRIORITY BITS
000016  010367  000144                  MOV     R3,INTPRI       ;STORE COMPLEMENTED PRIORITY
000022  012701  000020                  MOV     #INTSIZ,R1      ;# OF BYTES NEED FROM EXEC
000026                                  CALL    $ALOCB          ;ASK FOR THEM
000032  103447                          BCS     30$             ;NONE AVAILABLE
000034  010004                          MOV     R0,R4           ;SAVE THE BUFFER ADDRESS
000036  010005                          MOV     R0,R5
000040  012701  000162'                 MOV     #INTAD,R1       ;LOCATION FROM WHICH TO GET INSTRUCTIONS
000044  012125                  10$:    MOV     (R1)+,(R5)+     ;MOVE A WORD
000046  020127  000200'                 CMP     R1,#INTEND      ;LAST WORD MOVED?
000052  103774                          BLO     10$             ;NO
000054  016715  000000G                 MOV     $TKTCB,(R5)     ;STORE ' P ADDRESS IN LAST WORD
```

```
INTHND  MACRO M1104  06-DEC-77 08:24  PAGE 4-1
ASSEMBLE FOR USE WITH AN UNMAPPED SYSTEM

    116 000060  011600                          MOV     @SP,R0          ;RESTORE VECTOR ADDRESS
    117 000062  013746  177776                   MOV     @#PS,-(SP)      ;SAVE THE PROCESSOR STATUS
    118 000066  112737  000340  177776           MOVB    #340,@#PS       ;;;INHIBIT INTERRUPTS
    119 000074  021027  000000G                  CMP     (R0),#$NONSI    ;;;IS VECTOR CONTENTS
    120                                                                  ;;;ADDR OF NONSENSE INTERUPT RTN?
    121 000100  001406                           BEQ     15$             ;;;
    122 000102  021027  000000G                  CMP     (R0),#$NS0      ;;;CHECK ERRLOG LOCATIONS
    123 000106  002412                           BLT     20$             ;;;IF NO ERRLOG ARE SAME AS NONSI
    124 000110  021027  000000G                  CMP     (R0),#$NS7      ;;;
    125 000114  003007                           BGT     20$             ;;;
    126 000116                          15$:
    127 000116  010420                           MOV     R4,(R0)+        ;;;STORE BUFFER ADDR IN VECTOR
    128 000120  012637  177776                   MOV     (SP)+,@#PS      ;;;ENABLE INTERRUPTS
    129 000124  000241                           CLC                     ;CLEAR CARRY
    130 000126  004567  000000G                  JSR     R5,C.SRES       ;RESTORE THE REGISTERS
    131 000132  000207                           RTS     PC              ;AND RETURN TO CALLER
    132 000134  012637  177776          20$:     MOV     (SP)+,@#PS      ;;;ENABLE INTERRUPTS
    133 000140  010400                           MOV     R4,R0           ;GET NODE ADDRESS
    134 000142  012701  000020                   MOV     #INTSIZ,R1      ;AND ITS SIZE
    135 000146                                   CALL    $DEACB          ;AND RETURN IT
    136 000152  004567  000000G          30$:     JSR     R5,C.SRES       ;RESTORE REGISTERS
    137 000156  000261                           SEC                     ;SET CARRY TO INDICATE ERROR
    138 000160  000207                           RTS     PC              ;AND RETURN TO CALLER
    139                                 ;
    140                                 ;+
    141                                 ; **-INTAD-INTERRUPT SERVICE CODE
    142                                 ;
    143                                 ; THIS ROUTINE IS PUT INTO AN EXEC BUFFER BY ..CINT AND ITS ADDRESS
    144                                 ; (OF THE BUFFER) IS PLACED IN THE DEVICE VECTOR.
    145                                 ; IN AN UNMAPPED SYSTEM IT TRANSFERS TO THE USERS INTERRUPT
    146                                 ; SERVICE ROUTINE AND THEN TRANSFERS TO AN RSX-11M ROUTINE TO DISMISS
    147                                 ; THE INTERRUPT.
    148                                 ; IN A MAPPED SYSTEM, KERNEL APR4 IS SET TO POINT TO THE ADDRESS IN
    149                                 ; APR4 OF THE USER WHO DID THE CONNECT TO INTERRUPT.  IN THIS MANNER
    150                                 ; THE USER'S ISR CODE IS MAPPED INTO EXEC ADDRESSING SPACE.
    151                                 ; THIS ROUTINE MUST BE IN THE EXEC (I.E. AN EXEC BUFFER) TO GET
    152                                 ; CONTROL OF THE INTERRUPT AND CHANGE THE MAPPING.
    153                                 ;-
    154                                 ;
    155 000162  004537  000000G         INTAD:  JSR     R5,@#$INTSV     ;SWITCH STACKS
    156 000166  000000                  INTPRI: .WORD   0               ;PRIORITY (COMPLEMENTED)
    157                                         .IF DF M$$MGE
    158                                         MOV     @KISAR,-(SP)    ;SAVE KERNEL PAR
    159                                         MOV     #0,@KISAR       ;SET IT TO CONTENTS OF USER PAR
    160                                         I.KA=.-INTAD-4  ;POINT TO 0 WORD
    161                                         .ENDC
    162 000170  004737  000000                  JSR     PC,@#0          ;GO TO ISR
    163         000010                          INTJSR=.-2-INTAD        ;POINT TO 0 WORD
    164                                         .IF DF M$$MGE
    165                                         MOV     (SP)+,@KISAR    ;RESTORE KERNEL PAR
    166                                         .ENDC
    167 000174  000137  000000G                 JMP     @#$INTXT        ;SWITCH STACKS BACK
    168                                         .IF DF M$$MGE
    169                                 UISAR:  .WORD   UISAR5          ;
    170                                 KISAR:  .WORD   KISAR5          ;
    171                                         .ENDC
    172 000200  000020                  INTEND: INTSIZ=.-INTAD+2        ;LEAVE ROOM FOR TCB ADDRESS
```

```
                                        ;
                                        ;+
                                        ; **-..DINT-DISCONNECT FROM DEVICE VECTOR
                                        ;
                                        ; THIS ROUTINE IS CALLED TO DISCONNECT THE USER FROM THE DEVICE VECTOR
                                        ;
                                        ; INPUTS:
                                        ;
                                        ;       R0 = INTERRUPT TRAP ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;
                                        ; THE CONTENTS OF THE INTERRUPT TRAP ADDRESS (THE VECTOR) ARE RESET
                                        ; TO THE NONSENSE INTERRUPT ROUTINE ($NONSI) AND THE BUFFER (POINTED
                                        ; TO BY THE OLD CONTENTS OF THE VECTOR) IS RETURNED.  A CHECK
                                        ; IS MADE TO SEE IF THE TCB ADDRESS OF THE USER MAKING THE CALL IS THE
                                        ; SAME AS THE ONE WHICH DID THE CONNECT.  (THE TCB ADDR IS SAVED
                                        ; IN THE LAST WORD OF THE BUFFER BY ..CINT)  IF THEY DO NOT MATCH
                                        ; THE DISCONNECT IS NOT DONE BUT NO ERROR INDICATION IS GIVEN
                                        ;-
                                        ;
000200  010046                  ..DINT::  MOV     R0,-(SP)        ;SAVE R0
000202  010146                            MOV     R1,-(SP)        ;AND R1
000204  013746  177776                     MOV     @#PS,-(SP)      ;SAVE THE PROCESSOR STATUS
000210  112737  000340  177776             MOVB    #340,@#PS       ;;;INHIBIT INTERRUPTS
000216  011001                            MOV     @R0,R1          ;;;GET INTERRUPT ADDR FROM VECTOR
000220  026167  000016  000000G           CMP     INTSIZ-2(R1),$TKTCB ;;;DOES TCB ADDR OF CURRENT TASK
                                                                  ;;;MATCH THAT OF THE TASK THAT DID
                                                                  ;;;THE ORIGINAL CONNECT TO INTERRUPT?
000226  001007                            BNE     10$             ;;;NO
000230  012710  000000G                   MOV     #$NONSI,(R0)    ;;;PUT ADDR OF NONSENSE INTERRUPT RTN
                                                                  ;;;IN THE VECTOR
000234  010100                            MOV     R1,R0           ;;;GET NODE ADDRESS
000236  012701  000020                    MOV     #INTSIZ,R1      ;;;AND SIZE IN BYTES
000242                                    CALL    $DEACB          ;;;DEALOCATE THE BUFFER
000246  012637  177776          10$:      MOV     (SP)+,@#PS      ;;;RESTORE THE STATUS
000252  012601                            MOV     (SP)+,R1        ;RESTORE R1
000254  012600                            MOV     (SP)+,R0        ;AND R0
000256  000207                            RTS     PC              ;AND RETURN
        000001                            .END
```

```
INTHND  MACRO M1104  06-DEC-77 08:24  PAGE 5-1
SYMBOL TABLE

C.SREG= ****** GX        INTPRI  000166R     002 RSX    = 000001          $INTSV= ****** GX        $NS7  = ****** GX
C.SRES= ****** GX        INTSIZ= 000020         RSX11M= 000001          $INTXT= ****** GX        $TKTCB= ****** GX
INTAD   000162R     002 NRCVSF= 000000         TAGNUM= 000000          $NONSI= ****** GX        ..CINT  000000RG    002
INTEND  000200R     002 NTRNSF= 000002         $ALOCB= ****** GX       $NSO  = ****** GX        ..DINT  000200RG    002
INTJSR= 000010          PS    = 177776         $DEACB= ****** GX

.  ABS.   000000     000
          000000     001
MAIN      000260     002
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:   522 WORDS  ( 3 PAGES)
DYNAMIC MEMORY:   2810 WORDS  ( 10 PAGES)
ELAPSED TIME:   00:00:16
 256,020 INTHND.UMP;27, 256,030 INTHNDU/CRF/-SP=UAPRXM,UNMAP,INTHND
```

INTHNDU CREATED BY MACRO ON 6-DEC-77 AT 08:24 PAGE 1

SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES							
C.SREG	= ******	GX	4-92							
C.SRES	= ******	GX	4-130	4-136						
INTAD	000162	R	*4-102	4-111	#4-155	4-163	4-172			
INTEND	000200	R	4-113	#4-172						
INTJSR	= 000010		*4-102	#4-163						
INTPRI	000166	R	*4-105	#4-156						
INTSIZ	= 000020		4-106	4-134	#4-172	5-200	5-207			
M$$MGE	= ******		3-2	3-46	4-93	4-157	4-164	4-168		
NRCVSF	= 000000		#1-5							
NTRNSF	= 000002		#1-4							
PS	= 177776		#3-45	4-117	4-118	4-128	4-132	5-197	5-198	5-209
RSX	= 000001		#1-2							
RSX11M	= 000001		#1-3							
TAGNUM	= 000000		#1-8							
$ALOCB	= ******	GX	4-107							
$DEACB	= ******	GX	4-135	5-208						
$INTSV	= ******	GX	4-155							
$INTXT	= ******	GX	4-167							
$NONSI	= ******	GX	4-119	5-204						
$NSO	= ******	GX	4-122							
$NS7	= ******	GX	4-124							
$TKTCB	= ******	GX	4-115	5-200						
..CINT	000000	RG	#4-92							
..DINT	000200	RG	#5-195							

MACRO NAME	REFERENCES
CALL	#3-54 4-107 4-135 5-208
ERROR	#1-9
ERROPH	#1-16
ERRORM	#1-32
ERRORP	#1-47
FATAL	#1-53

.MAIN. MACRO M1104 06-DEC-77 08:23
TABLE OF CONTENTS

2- 1 ASSEMBLED FOR USE WITH A MAPPED SYSTEM

INTHND MACRO M1104 06-DEC-77 08:23 PAGE 1

```
                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM


                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
INTHND  MACRO M1104  06-DEC-77 08:23  PAGE 2
ASSEMBLED FOR USE WITH A MAPPED SYSTEM


                                        .SBTTL  ASSEMBLED FOR USE WITH A MAPPED SYSTEM
                                ;
000001                          M$$MGE=1                ;MEMORY MANAGEMENT
```

```
INTHND  MACRO M1104  06-DEC-77 08:23  PAGE 3
ASSEMBLED FOR USE WITH A MAPPED SYSTEM


                                        .TITLE  INTHND
                                        .IF DF  M$$MGE
                                        .IDENT /0003M/
                                        .IFF
                                        .IDENT  /0003U/
                                        .ENDC
                                ;
                                ; COPYRIGHT (C)  1975, 1976 ,1977
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                                ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                                ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                                ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                                ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                                ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                                ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                                ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                                ;
                                ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                                ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                                ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                                ;
                                ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                                ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                                ; DEC.
                                ;
                                ; VERSION 0
                                ;
                                ; ROBERT H.  ROSENBAUM  30-JULY-74
                                ;
                                ; LAST MODIFICATION ON 24-JUNE-75
                                ;
                                ; VERSION 1
                                ;
                                ; CLARISE E.  PATTON  SEPT 1977
                                ;
                                ; MODIFIED TO ACCEPT EITHER APR4 OR APR5
                                ;
                                ;
                                ; INTERRUPT RELATED ROUTINES FOR RSX-11M
                                ;
                                ; REGISTER DEFINITIONS
                                ;
  177776                                PS=177776
                                        .IF DF M$$MGE
  177650                                UISAR4=177650
  172350                                KISAR4=172350
  177652                                UISAR5=177652
  172352                                KISAR5=172352
                                        .ENDC
                                ;
                                ; MACRO CALLS
                                        .MCALL  CALL
                                ;
000000                                  .PSECT  MAIN
                                ;
```

```
INTHND  MACRO M1104  06-DEC-77 08:23  PAGE 4
ASSEMBLED FOR USE WITH A MAPPED SYSTEM


                                ;+
                                ; **-..CINT-CONNECT TO DEVICE VECTOR
                                ;
                                ; THIS ROUTINE IS CALLED TO ENABLE A USER TASK TO HANDLE INTERRUPTS
                                ; FROM A PARTICULAR DEVICE (SIMULATES THE RSX-11D EXEC ROUTINE
                                ; BY THE SAME NAME)
                                ;
                                ; INPUTS:
                                ;
                                ;       R0 -- INTERRUPT TRAP ADDRESS
                                ;       R1 -- ENTRY POINT OF INTERRUPT SERVICE ROUTINE
                                ;             (ADDRESS MUST BE WITHIN USER APR'S RANGE)
                                ;       R3 -- BITS 5 - 7 PRESCRIBE THE PRIORITY
                                ;
                                ; CALLING SEQUENCE:
                                ;
                                ;       JSR     PC,..CINT
                                ;
                                ; OUTPUTS:
                                ;
                                ;       THIS ROUTINE WILL CHECK TO SEE IF IT IS TO USE APR4 (APR5 IS
                                ;       DEFAULT), THEN WILL REQUEST A BUFFER FROM THE EXEC, STORE THE BUFFER
                                ;       ADDR IN THE VECTOR ADDR SUPPLIED BY THE CALLER, MOVE THE INSTRUCTIONS
                                ;       STARTING AT "INTAD" THROUGH "INTEND" INTO THE BUFFER FILLING IN
                                ;       THE ADDR CONTAINED IN USER APR4 AND THE USER'S ISR ADDR
                                ;       AND FINALLY RETURN TO THE USER WITH ALL REGISTERS UNCHANGED.
                                ;       THE "C" BIT WILL BE SET IF:
                                ;       1) THE VECTOR IS IN USE (I.E. THE ADDR IN THE VECTOR IS NOT
                                ;          THAT OF THE NONSENSE INTERRUPT ROUTINE ($NONSI) OR
                                ;       2) NO BUFFERS ARE AVAILABLE
                                ;
                                ;-
                                ;
000000  004567  000000G        ..CINT::JSR     R5,C.SREG       ;SAVE THE REGISTERS
                                .IF DF M$$MGE                   ;IF MEMORY MANGEMENT
000004  022727  000000G 120000          CMP     #STKSET,#120000         ;WHICH REGISTER USED?
000012  003006                          BGT     5$                      ;REGISTER 5
000014  012767  177650  000226          MOV     #UISAR4,UISAR   ;MOVE IN REGISTER 4
000022  012767  172350  000222          MOV     #KISAR4,KISAR           ;
000030                          5$:
000030  017767  000214  000172          MOV     @UISAR,INTAD+1.KA       ;PUTS CONTENTS OF PAR INTO
                                                                        ;MOVE INSTRUCTION
                                .ENDC
000036  010167  000174                  MOV     R1,INTAD+INTJSR ;STORE ADDR INTO JSR INTS
000042  005103                          COM     R3              ;COMPLEMENT THE PRIORITY
000044  042703  177437                  BIC     #177437,R3      ;CLEAR ALL BUT PRIORITY BITS
000050  010367  000144                  MOV     R3,INTPRI       ;STORE COMPLEMENTED PRIORITY
000054  012701  000042                  MOV     #INTSIZ,R1      ;# OF BYTES NEED FROM EXEC
000060                                  CALL    $ALOCB          ;ASK FOR THEM
000064  103447                          BCS     30$             ;NONE AVAILABLE
000066  010004                          MOV     R0,R4           ;SAVE THE BUFFER ADDRESS
000070  010005                          MOV     R0,R5
000072  012701  000214'                 MOV     #INTAD,R1       ;LOCATION FROM WHICH TO GET INSTRUCTIONS
000076  012125                  10$:    MOV     (R1)+,(R5)+     ;MOVE A WORD
000100  020127  000254'                 CMP     R1,#INTEND      ;LAST WORD MOVED?
000104  103774                          BLO     10$             ;NO
000106  016715  000000G                 MOV     $TKTCB,(R5)     ;STORE TCB ADDRESS IN LAST WORD
```

```
INTHND  MACRO M1104  06-DEC-77 08:23  PAGE 4-1
ASSEMBLED FOR USE WITH A MAPPED SYSTEM

    116 000112  011600                          MOV     @SP,R0          ;RESTORE VECTOR ADDRESS
    117 000114  013746  177776                  MOV     @#PS,-(SP)      ;SAVE THE PROCESSOR STATUS
    118 000120  112737  000340  177776          MOVB    #340,@#PS       ;;;INHIBIT INTERRUPTS
    119 000126  021027  000000G                 CMP     (R0),#$NONSI    ;;;IS VECTOR CONTENTS
    120                                                                 ;;;ADDR OF NONSENSE INTERUPT RTN?
    121 000132  001406                          BEQ     15$             ;;;
    122 000134  021027  000000G                 CMP     (R0),#$NS0      ;;;CHECK ERRLOG LOCATIONS
    123 000140  002412                          BLT     20$             ;;;IF NO ERRLOG ARE SAME AS NONSI
    124 000142  021027  000000G                 CMP     (R0),#$NS7      ;;;
    125 000146  003007                          BGT     20$             ;;;
    126 000150                          15$:
    127 000150  010420                          MOV     R4,(R0)+        ;;;STORE BUFFER ADDR IN VECTOR
    128 000152  012637  177776                  MOV     (SP)+,@#PS      ;;;ENABLE INTERRUPTS
    129 000156  000241                          CLC                     ;CLEAR CARRY
    130 000160  004567  000000G                 JSR     R5,C.SRES       ;RESTORE THE REGISTERS
    131 000164  000207                          RTS     PC              ;AND RETURN TO CALLER
    132 000166  012637  177776          20$:    MOV     (SP)+,@#PS      ;;;ENABLE INTERRUPTS
    133 000172  010400                          MOV     R4,R0           ;GET NODE ADDRESS
    134 000174  012701  000042                  MOV     #INTSIZ,R1      ;AND ITS SIZE
    135 000200                                  CALL    $DEACB          ;AND RETURN IT
    136 000204  004567  000000G         30$:    JSR     R5,C.SRES       ;RESTORE REGISTERS
    137 000210  000261                          SEC                     ;SET CARRY TO INDICATE ERROR
    138 000212  000207                          RTS     PC              ;AND RETURN TO CALLER
    139                                 ;
    140                                 ;+
    141                                 ; **-INTAD-INTERRUPT SERVICE CODE
    142                                 ;
    143                                 ; THIS ROUTINE IS PUT INTO AN EXEC BUFFER BY .CINT AND ITS ADDRESS
    144                                 ; (OF THE BUFFER) IS PLACED IN THE DEVICE VECTOR.
    145                                 ; IN AN UNMAPPED SYSTEM IT TRANSFERS TO THE USERS INTERRUPT
    146                                 ; SERVICE ROUTINE AND THEN TRANSFERS TO AN RSX-11M ROUTINE TO DISMISS
    147                                 ; THE INTERRUPT.
    148                                 ; IN A MAPPED SYSTEM, KERNEL APR4 IS SET TO POINT TO THE ADDRESS IN
    149                                 ; APR4 OF THE USER WHO DID THE CONNECT TO INTERRUPT.  IN THIS MANNER
    150                                 ; THE USER'S ISR CODE IS MAPPED INTO EXEC ADDRESSING SPACE.
    151                                 ; THIS ROUTINE MUST BE IN THE EXEC (I.E. AN EXEC BUFFER) TO GET
    152                                 ; CONTROL OF THE INTERRUPT AND CHANGE THE MAPPING.
    153                                 ;-
    154                                 ;
    155 000214  004537  000000G         INTAD:  JSR     R5,@#$INTSV     ;SWITCH STACKS
    156 000220  000000                  INTPRI: .WORD   0               ;PRIORITY (COMPLEMENTED)
    157                                         .IF DF M$$MGE
    158 000222  017746  000024                  MOV     @KISAR,-(SP)    ;SAVE KERNEL PAR
    159 000226  012777  000000  000016          MOV     #0,@KISAR       ;SET IT TO CONTENTS OF USER PAR
    160         000014                          I.KA=.-INTAD-4  ;POINT TO 0 WORD
    161                                         .ENDC
    162 000234  004737  000000                  JSR     PC,@#0          ;GO TO ISR
    163         000022                          INTJSR=.-2-INTAD        ;POINT TO 0 WORD
    164                                         .IF DF M$$MGE
    165 000240  012677  000006                  MOV     (SP)+,@KISAR    ;RESTORE KERNEL PAR
    166                                         .ENDC
    167 000244  000137  000000G                 JMP     @#$INTXT        ;SWITCH STACKS BACK
    168                                         .IF DF M$$MGE
    169 000250  177652                  UISAR:  .WORD   UISAR5          ;
    170 000252  172352                  KISAR:  .WORD   KISAR5          ;
    171                                         .ENDC
    172 000254  000042                  INTEND: INTSIZ=.-INTAD+2        ;LEAVE ROOM FOR TCB ADDRESS
```

```
INTHND  MACRO M1104  06-DEC-77 08:23  PAGE 5
ASSEMBLED FOR USE WITH A MAPPED SYSTEM

                                        ;
                                        ;+
                                        ; **-..DINT-DISCONNECT FROM DEVICE VECTOR
                                        ;
                                        ; THIS ROUTINE IS CALLED TO DISCONNECT THE USER FROM THE DEVICE VECTOR
                                        ;
                                        ; INPUTS:
                                        ;
                                        ;       R0 = INTERRUPT TRAP ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;
                                        ; THE CONTENTS OF THE INTERRUPT TRAP ADDRESS (THE VECTOR) ARE RESET
                                        ; TO THE NONSENSE INTERRUPT ROUTINE ($NONSI) AND THE BUFFER (POINTED
                                        ; TO BY THE OLD CONTENTS OF THE VECTOR) IS RETURNED.  A CHECK
                                        ; IS MADE TO SEE IF THE TCB ADDRESS OF THE USER MAKING THE CALL IS THE
                                        ; SAME AS THE ONE WHICH DID THE CONNECT.  (THE TCB ADDR IS SAVED
                                        ; IN THE LAST WORD OF THE BUFFER BY ..CINT)  IF THEY DO NOT MATCH
                                        ; THE DISCONNECT IS NOT DONE BUT NO ERROR INDICATION IS GIVEN
                                        ;-
                                        ;
000254  010046                          ..DINT::MOV     R0,-(SP)        ;SAVE R0
000256  010146                                  MOV     R1,-(SP)        ;AND R1
000260  013746  177776                          MOV     @#PS,-(SP)      ;SAVE THE PROCESSOR STATUS
000264  112737  000340  177776                  MOVB    #340,@#PS       ;;;INHIBIT INTERRUPTS
000272  011001                                  MOV     @R0,R1          ;;;GET INTERRUPT ADDR FROM VECTOR
000274  026167  000040  000000G                CMP     INTSIZ-2(R1),$TKTCB ;;;DOES TCB ADDR OF CURRENT TASK
                                                                        ;;;MATCH THAT OF THE TASK THAT DID
                                                                        ;;;THE ORIGINAL CONNECT TO INTERRUPT?
000302  001007                                  BNE     10$             ;;;NO
000304  012710  000000G                         MOV     #$NONSI,(R0)    ;;;PUT ADDR OF NONSENSE INTERRUPT RTN
                                                                        ;;;IN THE VECTOR
000310  010100                                  MOV     R1,R0           ;;;GET NODE ADDRESS
000312  012701  000042                          MOV     #INTSIZ,R1      ;;;AND SIZE IN BYTES
000316                                          CALL    $DEACB          ;;;DEALOCATE THE BUFFER
000322  012637  177776          10$:            MOV     (SP)+,@#PS      ;;;RESTORE THE STATUS
000326  012601                                  MOV     (SP)+,R1        ;RESTORE R1
000330  012600                                  MOV     (SP)+,R0        ;AND R0
000332  000207                                  RTS     PC              ;AND RETURN
        000001                                  .END
```

```
INTHND  MACRO M1104  06-DEC-77 08:23  PAGE 5-1
SYMBOL TABLE

C.SREG= ****** GX        I.KA  = 000014          PS    = 177776          UISAR4= 177650          $NONSI= ****** GX
C.SRES= ****** GX        KISAR   000252R  002 RSX   = 000001          UISAR5= 177652          $NSO  = ****** GX
INTAD   000214R  002 KISAR4= 172350          RSX11M= 000001          $ALOCB= ****** GX       $NS7  = ****** GX
INTEND  000254R  002 KISAR5= 172352          STKSET= ****** GX       $DEACB= ****** GX       $TKTCB= ****** GX
INTJSR= 000022           M$$MGE= 000001      TAGNUM= 000000          $INTSV= ****** GX       ..CINT  000000RG    002
INTPRI  000220R  002 NRCVSF= 000000          UISAR   000250R  002 $INTXT= ****** GX       ..DINT  000254RG    002
INTSIZ= 000042           NTRNSF= 000002

.  ABS.   000000    000
          000000    001
MAIN      000334    002
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:   577 WORDS  ( 3 PAGES)
DYNAMIC MEMORY:   2810 WORDS  ( 10 PAGES)
ELAPSED TIME:   00:00:21
 256 020 INTHND.MAP;27, 256,030 INTHNDM/CRF/-SP=UAPRXM,MAP,INTHND
```

SYMBOL	VALUE		REFERENCES							
C.SREG	= ******	GX	4-92							
C.SRES	= ******	GX	4-130	4-136						
INTAD	000214	R	*4-99	*4-102	4-111	#4-155	*4-160	4-163	4-172	
INTEND	000254	R	4-113	#4-172						
INTJSR	= 000022		*4-102	#4-163						
INTPRI	000220	R	*4-105	#4-156						
INTSIZ	= 000042		4-106	4-134	#4-172	5-200	5-207			
I.KA	= 000014		*4-99	#4-160						
KISAR	000252	R	*4-97	4-158	4-159	4-165	#4-170			
KISAR4	= 172350		#3-48	4-97						
KISAR5	= 172352		#3-50	4-170						
MSGMGE	= 000001		#2-3	3-2	3-46	4-93	4-157	4-164	4-168	
NRCVSF	= 000000		#1-5							
NTRNSF	= 000002		#1-4							
PS	= 177776		#3-45	4-117	4-118	4-128	4-132	5-197	5-198	5-209
RSX	= 000001		#1-2							
RSX11M	= 000001		#1-3							
STKSET	= ******	GX	4-94							
TAGNUM	= 000000		#1-8							
UISAR	000250	R	*4-96	4-99	#4-169					
UISAR4	= 177650		#3-47	4-96						
UISAR5	= 177652		#3-49	4-169						
$ALOCB	= ******	GX	4-107							
$DEACB	= ******	GX	4-135	5-208						
$INTSV	= ******	GX	4-155							
$INTXT	= ******	GX	4-167							
$NONSI	= ******	GX	4-119	5-204						
$NSO	= ******	GX	4-122							
$NS7	= ******	GX	4-124							
$TKTCB	= ******	GX	4-115	5-200						
..CINT	000000	RG	#4-92							
..DINT	000254	RG	#5-195							

MACRO NAME	REFERENCES			
CALL	#3-54	4-107	4-135	5-208
ERROR	#1-9			
ERRORK	#1-16			
ERRORM	#1-32			
ERRORP	#1-47			
FATAL	#1-53			

```
                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM


                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                                .TITLE  LOOKUP
                                .IDENT  /00004/
                        ;
                        ; COPYRIGHT (C)  1975, 1976, 1977
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                        ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                        ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                        ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                        ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                        ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                        ;
                        ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ; DEC.
                        ;


                        ; VERSION 0
                        ;
                        ; ROBERT H. ROSENBAUM 2-AUGUST-74
                        ; LAST MODIFICATION ON 20-FEBRUARY-75
                        ;
                        ; FIND ALL FILES FOR INPUT (TRANSMISSION COMMAND)
                        ;
000000                          .PSECT MAIN
                        ;
                        ; MACRO CALLS
                                .MCALL  CSI$2,CSI$SW,CSI$ND,FDOFF$,FCSBT$,CALL
000000                  FDOFF$  DEF$L   ;DEFINE FCS OFFSETS LOCALLY
000000                  FCSBT$  DEF$L   ;DEFINE FCS BITS LOCALLY
                        ;
                        ;+
                        ; INPUTS:
                        ;       R2 = ADDRESS TO STORE PARSED COMMAND STRING
                        ;
                        ; OUTPUTS:
                        ;       FILE NAME BLOCK INFORMATION IS PLACED IN AREA POINTED
                        ;       TO BY R2 UPON ENTRY.
                        ;       IF AN ERROR CONDITION IS DETECTED, A MESSAGE WILL BE QUEUED
                        ;       AND CARRY WILL BE SET.
                        ;-
                                .IF DF  RSX
                                .ENABL  LSB             ;ENABLE LOCAL SYMBOL BLOCK
000000  010201          LUKUP:: MOV     R2,R1           ;SAVE AREA FOR FNB
000002  005067  000000G 10$:    CLR     SWREN           ;CLEAR END SWITCH
000006  005067  000000G        CLR     SWTBN           ;CLEAR  BINARY SWITCH
000012                          CSI$2   #FILDES,INPUT,#SWTAB
000034  103474                BCS     40$
000036  012700  000000G        MOV     #FDBIN,R0       ;FDB BLOCK
000042  012702  000006G        MOV     #FILDES+C.DSDS,R2
```

```
                                                MOV     #DFBLK,R3       ;DEFAULT FILE NAME BLOCK
                                                CALL    .PARSE          ;PARSE IT
                                                BCS     50$
                                                BITB    #NB.SVR!NB.SNM!NB.STP,N.STAT(R1)        ;ANY WILD CARDS?
                                                BNE     35$             ;YES ILLEGAL
                                                CALL    .FIND           ;AND FIND IT
                                                BCS     30$             ;AN ERROR
                                                ADD     #S.FNB,R1
                                                MOV     #SWTBN,R3
                                                MOV     #XMTSWT/2,R0    ;# OF WORDS OF XMIT SWITCHES
                                        15$:    MOV     (R3)+,(R1)+     ;MOVE TO TRANSMIT BLOCK AREA
                                                DEC     R0              ;ONE LESS SWITCH
                                                BNE     15$             ;MOVE THE NEXT ONE
                                                BITB    #CS.MOR,C.STAT+FILDES
                                                BEQ     20$
                                                CMP     R1,#TRNSE       ;TOO MANY FILES
                                                BLO     10$             ;NO, LOOK FOR NEXT FILE
                                                CLR     @TRNSBLK        ;YES, CLEAR WHAT HAS BEEN DONE
                                                ERRORP  <TOO MANY XMIT FILES>   ;AND INFORM OPERATOR OF ERROR
                                                BR      60$
                                        20$:    CLR     (R1)
                                                CLC                     ;CLEAR CARRY
                                                RTS     PC
                                        30$:    JSR     PC,CSIERR               ;PRINT STRING IN ERROR
                                                ERRORP  <NO SUCH FILE>
                                                BR      60$
                                        35$:    ERRORP  <WILD CARD FND>
                                                BR      60$
                                        40$:    JSR     PC,CSIERR               ;PRINT STRING IN ERROR
                                                ERRORP  <ILL SWITCH>
                                                BR      60$
                                        50$:    ERRORP  <PARSE FAIL>
                                        60$:    CLR     @TRNSBLK
                                                SEC                     ;SET CARRY
                                                RTS     PC              ;AND RETURN
                                                .DSABL  LSB
```

```
                                        ;+
                                        ; COMMAND STRING INTERPRETER'S SWITCH TABLE FOR TRANSMIT COMMANDS
                                        ;-
000274                                  SWTAB:
000274                                          CSI$SW  B1,200,SWTBN
000304                                          CSI$SW  EN,200,SWREN
000314                                          CSI$ND
                                                .ENDC
        000001                                  .END
```

```
LOOKUP  MACRO M1104  06-DEC-77 08:25  PAGE 3-1
SYMBOL TABLE

CHR$$ = 000116          FA.WCK= 020000          F.ALOC= 000040          F.RSIZ= 000002          R.FIX = 000001
CH.AND= 000001          FA.WRT= 000002          F.BBFS= 000062          F.RTYP= 000000          R.SEQ = 000003
CNT$$ = 000002          FDBIN = ****** GX       F.BDB = 000070          F.SEQN= 000100          R.VAR = 000002
CSIERR= ****** GX       FD.BLK= 000010          F.BGBC= 000057          F.SPDV= 000072          SIZ$$ = 000002
CS.DIF= 000002          FD.CCL= 000002          F.BKDN= 000026          F.SPUN= 000074          SWREN = ****** GX
CS.DVF= 000004          FD.COM= 020000          F.BKDS= 000020          F.STBK= 000036          SWTAB   000274R     002
CS.EQU= 000040          FD.CR = 000002          F.BKEF= 000050          F.UNIT= 000136          SWTBN = ****** GX
CS.INP= 000001          FD.DIR= 000010          F.BKP1= 000051          F.URBD= 000020          S.FATT= 000016
CS.MOR= 000020          FD.FTN= 000001          F.BKST= 000024          F.VBN = 000064          S.FDB = 000140
CS.NMF= 000001          FD.F11= 040000          F.BKVB= 000064          F.VBSZ= 000060          S.FNAM= 000006
CS.OUT= 000002          FD.INS= 000010          F.CHR = 000075          LUKUP   000000RG    002 S.FNB = 000036
CS.WLD= 000010          FD.ISP= 002000          F.CNTG= 000034          NB.DEV= 000200          S.FNBW= 000017
C.CMLD= 000002          FD.MNT= 100000          F.DFNB= 000046          NB.DIR= 000100          S.FNTY= 000004
C.DEVD= 000006          FD.OSP= 004000          F.DSPT= 000044          NB.NAM= 000004          S.FTYP= 000002
C.DIRD= 000012          FD.PLC= 000004          F.DVNM= 000134          NB.SD1= 000400          S.NFEN= 000020
C.DSDS= 000006          FD.PSE= 010000          F.EFBK= 000010          NB.SD2= 001000          TAGE1   000023R     003
C.FILD= 000016          FD.RAH= 000001          F.EFN = 000050          NB.SNM= 000040          TAGE2   000037R     003
C.MKW1= 000024          FD.RAN= 000002          F.EOBB= 000032          NB.STP= 000020          TAGE3   000054R     003
C.MKW2= 000026          FD.REC= 000001          F.ERR = 000052          NB.SVR= 000010          TAGE4   000066R     003
C.SIZE= 000054          FD.RWM= 000001          F.FACC= 000043          NB.TYP= 000002          TAGE5   000100R     003
C.STAT= 000001          FD.SDI= 000020          F.FFBY= 000014          NB.VER= 000001          TAGNUM= 000005
C.SWAD= 000022          FD.SQD= 000040          F.FNAM= 000110          NRCVSF= 000000          TAG1    000000R     003
C.TYPR= 000000          FD.TTY= 000004          F.FNB = 000102          NTRNSF= 000002          TAG2    000023R     003
DFBLK = ****** GX       FD.WBH= 000002          F.FTYP= 000116          N.DID = 000024          TAG3    000037R     003
FA.APD= 000100          FF.CHR= 000005          F.FVER= 000120          N.DVNM= 000032          TAG4    000054R     003
FA.CRE= 000010          FF.NV = 000003          F.HIBK= 000004          N.FID = 000000          TAG5    000066R     003
FA.DLK= 001000          FF.POE= 000002          F.LUN = 000042          N.FNAM= 000006          TRNSBL= ****** GX
FA.ENB= 100000          FF.RWD= 000001          F.MBCT= 000054          N.FTYP= 000014          TRNSE = ****** GX
FA.EXC= 002000          FF.RWF= 000006          F.MBC1= 000055          N.FVER= 000016          TYPMS1= ****** GX
FA.EXT= 000004          FF.SPC= 000004          F.MBFG= 000056          N.NEXT= 000022          XMTSWT= ****** GX
FA.NSP= 000100          FILDES= ****** GX       F.NRBD= 000024          N.STAT= 000020          $$    = 000000
FA.POS= 010000          FO.APD= 000106          F.NREC= 000030          N.UNIT= 000034          .CSI2 = ****** G
FA.RD = 000001          FO.MFY= 000002          F.OVBS= 000030          PAR$$$= 000027          .FIND = ****** GX
FA.RWD= 004000          FO.RD = 000001          F.RACC= 000016          PROMPT= ****** GX       .PARSE= ****** GX
FA.SEQ= 040000          FO.UPD= 000006          F.RATT= 000001          RSX   = 000001          ...GBL= 000000
FA.SHR= 000040          FO.WRT= 000016          F.RCNM= 000034          RSX11M= 000001          ...TPC= 000140
FA.TMP= 000020          F.ACTL= 000076          F.RCTL= 000017

.  ABS.   000000    000
          000000    001
MAIN      000316    002
MESAG2    000100    003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:   3725 WORDS  ( 15 PAGES)
DYNAMIC MEMORY:   4922 WORDS  ( 18 PAGES)
ELAPSED TIME:   00:00:27
 256,020 LOOKUP;27, 256,030 LOOKUP/CRF/-SP=UAPRXM,LOOKUP
```

LOOKUP CREATED BY MACRO ON 6-DEC-77 AT 08:25 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES								
CHR$$	= 000116		#3-99	3-99	3-99	#3-99	3-99	3-99	#3-100	3-100	3-100
			#3-100	3-100	3-100						
CNT$$	= 000002		#3-99	3-99	#3-99	3-99	3-99	#3-99	3-99	#3-100	3-100
			#3-100	3-100	3-100	#3-100	3-100				
CSIERR	= ******	GX	2-81	2-86							
CS.INP	= 000001		2-54								
CS.MOR	= 000020		2-71								
C.DSDS	= 000006		2-57								
C.SIZE	= 000054		2-54								
C.STAT	= 000001		2-71								
C.SWAD	= 000022		*2-54								
DFBLK	= ******	GX	2-58								
FDBIN	= ******	GX	2-56								
FILDES	= ******	GX	2-54	2-54	2-57	2-71					
LUKUP	000000	RG	#2-51								
NB.SNM	= 000040		2-61								
NB.STP	= 000020		2-61								
NB.SVR	= 000010		2-61								
NRCVSF	= 000000		#1-5								
NTRNSF	= 000002		#1-4								
N.STAT	= 000020		2-61								
PAR$$$	= 000027		#2-54	2-54							
PROMPT	= ******	GX	2-76	2-82	2-84	2-87	2-89				
RSX	= 000001		#1-2	2-49							
RSX11M	= 000001		#1-3								
SIZ$$	= 000002		#3-99	3-99	3-99	3-99	#3-100	3-100	3-100	3-100	
SWREN	= ******	GX	*2-52	3-100							
SWTAB	000274	R	2-54	#3-98							
SWTBN	= ******	GX	*2-53	2-66	3-99						
S.FNB	= 000036		2-65								
TAGE1	000023	R	2-76	#2-76							
TAGE2	000037	R	2-82	#2-82							
TAGE3	000054	R	2-84	#2-84							
TAGE4	000066	R	2-87	#2-87							
TAGE5	000100	R	2-89	#2-89							
TAGNUM	= 000005		#1-8	2-76	#2-76	2-76	2-82	#2-82	2-82	2-84	#2-84
			2-84	2-87	#2-87	2-87	2-89	#2-89	2-89		
TAG1	000000	R	2-76	2-76	#2-76						
TAG2	000023	R	2-82	2-82	#2-82						
TAG3	000037	R	2-84	2-84	#2-84						
TAG4	000054	R	2-87	2-87	#2-87						
TAG5	000066	R	2-89	2-89	#2-89						
TRNSBL	= ******	GX	2-75	2-90							
TRNSE	= ******	GX	2-73								
TYPMS1	= ******	GX	2-76	2-82	2-84	2-87	2-89				
XMTSWT	= ******	GX	2-67								
$$	= 000000		#3-99	3-99	#3-99	3-99	#3-99	3-99	#3-99	3-99	#3-100
			3-100	#3-100	3-100	#3-100	3-100	#3-100	3-100		
.CSI2	= ******	G	2-54	2-54							
.FIND	= ******	GX	2-63								
.PARSE	= ******	GX	2-59								

LOOKUP CREATED BY MACRO ON 6-DEC-77 AT 08: 25
MACRO CROSS REFERENCE CREF

MACRO NAME	REFERENCES					
CALL	#2-35	#2-59	#2-63			
CSI$	#2-54					
CSI$ND	#2-35	#3-101				
CSI$SW	#2-35	#3-99	#3-100			
CSI$2	#2-35	#2-54				
ERROR	#1-9					
ERRORK	#1-16					
ERRORM	#1-32	#2-76	#2-82	#2-84	#2-87	#2-89
ERRORP	#1-47	2-76	2-82	2-84	2-87	2-89
FATAL	#1-53					
FCSBT$	#2-35	#2-37				
FDOFF$	#2-35	#2-36				
LDPO$	#2-54	#2-54				

```
                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM


                                ;
                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                                        .TITLE  MRKPNT
                                        .IDENT  /00006/
                                ;
                                ; COPYRIGHT (C)  1975, 1976, 1977
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                                ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                                ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                                ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                                ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                                ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                                ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                                ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                                ;
                                ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                                ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                                ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                                ;
                                ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                                ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                                ; DEC.
                                ;
                                ; V00-4 10/1/74
                                ;
                                ; ROUTINES TO SAVE AND RESORE PLACE IN FILE
                                ;
                                ; MACRO CALLS
                                        .MCALL  CALL,RETURN
                                ;
000000                                  .PSECT  MAIN
                                ;
                                ;+
                                ; **-POINT - POINT TO PLACE IN FILE
                                ; INPUTS:
                                ;       POINT1-3 - CORRESPOND TO R1-R3 SAVED AFTER .MARK
                                ;
                                ; OUTPUTS:
                                ;       FDB IS SET SO NEXT READ WILL RETREIVE THE RECORD AFTER THE
                                ;       ONE LAST READ BEFORE THE CALL TO MARK
                                ;-
                                        .ENABL  LSB
000000  004567  000000G       POINT:: JSR     R5,C.SREG       ;SAVE THE REGISTERS
000004  012700  000000G               MOV     #FDBIN,R0       ;FDB ADDRESS
000010  016701  000000G               MOV     POINT1,R1       ;RETRIEVE SAVED POINTERS
000014  016702  000000G               MOV     POINT2,R2       ; ...
000020  016703  000000G               MOV     POINT3,R3       ; ...
000024                                CALL    .POINT          ; CALL FCS TO SET UP FDB
000030  103010                        BCC     10$             ;BRANCH IF NO ERROR
000032                                ERROR   <POINT ERR>     ;ERROR WHILE TRYING TO POINT
000046  000261                        SEC                     ;SET THE CARRY FLAG
000050  000402                        BR      20$             ;AND RETURN
000052  105067  000000G       10$:    CLRB    ELATCH          ;ALLOW READS AGAIN
000056  004567  000000G       20$:    JSR     R5,C.SRES       ;RESTORE THE REGISTERS
000062                                RETURN                  ;AND RETURN
                                        .DSABL  LSB
```

```
                                        ;+
                                        ; **-MRKFIL - SAVE PLACE IN FILE
                                        ; INPUTS:
                                        ;       NONE
                                        ; OUTPUTS:
                                        ;       POINT1-3 CONTAIN R1-R3 AFTER CALL TO FCS ROUTINE .MARK
                                        ;-
000064  004567  000000G         MRKFIL::JSR     R5,C.SREG       ;SAVE THE REGISTERS
000070  152767  000200  000000G        BISB    #200,ELATCH     ;NO MORE READS
000076  012700  000000G                MOV     #FDBIN,R0       ;GET ADDRESS OF INPUT FDB
000102                                  CALL    .MARK           ;CALL FCS ROUTINE TO FIND PLACE IN FILE
000106  010167  000000G                MOV     R1,POINT1       ;SAVE INFO RETURNED
000112  010267  000000G                MOV     R2,POINT2       ; ...
000116  010367  000000G                MOV     R3,POINT3       ; ...
000122  152767  000040  000000G        BISB    #40,TXPEN       ;REMEMBER TO CONTINUE LATER
000130  004567  000000G                JSR     R5,C.SRES       ;RESTORE THE REGISTERS
000134                                  RETURN                  ;AND RETURN
                                        ;
        000001                          .END
```

```
MRKPNT  MACRO M1104  06-DEC-77 08:25  PAGE 3-1
SYMBOL TABLE

C.SREG= ****** GX        NRCVSF= 000000          POINT2= ****** GX       RSX11M= 000001          TXFEN = ****** GX
C.SRES= ****** GX        NTFNSF= 000002          POINT3= ****** GX       TAGE1   000011R     003 TYPMSG= ****** GX
ELATCH= ****** GX        POINT   000000RG    002 PROMPT= ****** GX       TAGNUM= 000001          .MARK = ****** GX
FDBIN = ****** GX        POINT1= ****** GX       RSX   = 000001          TAG1    000000R     003 .POINT= ****** GX
MRKFIL  000064RG    002

  ABS.  000000      000
        000000      001
MAIN    000136      002
MESAG2  000011      003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:   635 WORDS  ( 3 PAGES)
DYNAMIC MEMORY:   2810 WORDS  ( 10 PAGES)
ELAPSED TIME:   00:00:06
 256.020 MRKPNT;27, 256,030 MRKPNT/CRF/-SP=UAPRXM,MRKPNT
```

MRKPNT CREATED BY MACRO ON 6-DEC-77 AT 08:25
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES			
C.SREG	= ******	GX	2-43	3-65		
C.SRES	= ******	GX	2-54	3-73		
ELATCH	= ******	GX	*2-53	*3-66		
FDBIN	= ******	GX	2-44	3-67		
MRKFIL	000064	RG	#3-65			
NRCVSF	= 000000		#1-5			
NTRNSF	= 000002		#1-4			
POINT	000000	RG	#2-43			
POINT1	= ******	GX	2-45	*3-69		
POINT2	= ******	GX	2-46	*3-70		
POINT3	= ******	GX	2-47	*3-71		
PROMPT	= ******	GX	2-50			
RSX	= 000001		#1-2			
RSX11M	= 000001		#1-3			
TAGE1	000011	R	2-50	#2-50		
TAGNUM	= 000001		#1-8	2-50	#2-50	2-50
TAG1	000000	R	2-50	2-50	#2-50	
TXPEN	= ******	GX	*3-72			
TYPMSG	= ******	GX	2-50			
.MARK	= ******	GX	3-68			
.POINT	= ******	GX	2-48			

MACRO NAME	REFERENCES		
CALL	#2-29	2-48	3-68
ERROR	#1-9	2-50	
ERRORK	#1-16	2-50	
ERRORM	#1-32		
ERRORP	#1-47		
FATAL	#1-53		
RETURN	#2-29	2-55	3-74

COMTEX-MACROS MACRO M1104 06-DEC-77 08:24
TABLE OF CONTENTS

3- 21 COMTEX MACRO DEFINITIONS
5- 83 INITALIZE LINE TABLES

```
                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM


                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                        .TITLE  COMTEX-MACROS
                        .IDENT  /00002/
                ;
                ;COPYRIGHT 1974, 1975, 1976, 1977
                ;DIGITAL EQUIPMENT CORP.
                ;146 MAIN STREET        MAYNARD, MASS. 01754
                ;
                ;DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR
                ;RELIABILITY OF ITS SOFTWARE ON EQUIPMENT WHICH
                ;IS NOT SUPPLIED BY DIGITAL
                ;
                ; VERSION 0
                ;
                ; 23-SEPTEMBER-74
                ;
                ;VERSION 1
                ; SEP 1977 - CEP
                ; MODIFICATIONS FOR CUT CODE VERSION OF 2780
                ; LIBRARY OF COMTEX MACRO DEFINITIONS
```

```
                        .SBTTL  COMTEX MACRO DEFINITIONS
                ;
                        .MACRO  LINIT,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTINT
                        .ENDM
                ;
                        .MACRO  ASRBUF,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTPUT
                        .ENDM
                ;
                        .MACRO  PUTTC,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTPUT
                        .ENDM
                ;
                        .MACRO  PUTMC,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTPUT
                        .ENDM
                ;
                        .MACRO  GETS
                        JSR     PC.EMTGET
                        .ENDM
                ;
                        .MACRO  PUTTM,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTTIM
                        .ENDM
                ;
                        .MACRO  PUTD,PARLST
                        MOV     #PARLST,R1
                        JSR     PC.EMTPUT
                        .ENDM
```

```
INITAL  MACRO M1104  06-DEC-77 08:24  PAGE 4
COMTEX MACRO DEFINITIONS

                                        .TITLE  INITAL
                                        .IDENT /00013/
                                ;
                                ; COPYRIGHT (C) 1974,1975,1976,1977
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY
                                ; ON A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                                ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                                ; SOFTWARE, OR ANY COPIES THEREOF, MAY NOT BE PROVIDED OR
                                ; OTHERWISE MADE AVBAILABLE TO ANY OTHER PERSON EXCEPT
                                ; FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO THESE
                                ; LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE SOFTWARE
                                ; SHALL AT ALL TIMES REMAIN IN DEC.
                                ;
                                ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                                ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                                ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                                ;
                                ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                                ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                                ; DEC.
                                ;
                                ; BLR   7/08/75
                                ; MODIFIED FOR RT-11 V02-1 (F/B)         EDIT 5
                                ; MODIFIED FOR CUT CODE VERSION
                                ; CEP 8/77
                                ;
                                ; MODIFIED TOCLEAR SST VECTORS
                                ; CEP 9/77
                                ; MODIFIED TO NOT CLEAR CSR IF CALLED WHEN SOMEONE ELSE USING DEVICE
                                ; CEP 10/77
                                ;
                                ;
                                ; INITIALIZATION AND EXIT ROUTINES
                                ;
                                        .IFDF RSX
000000                                  .PSECT  MAIN
                                        .ENDC
                                        .IF DF RT11
                                        .CSECT  INITIA
                                        .GLOBL  CR.TIC          ;CRB TIMER IN TICS
                                        .ENDC
        177776                          PS=177776
                                ;
                                ; MACRO CALLS
                                        .IF DF  RSX
                                        .MCALL  DIR$,WTSE$S,EXIT$S,FINIT$,QIO$,GMCR$
                                        .IF NDF RSX11M
                                        .MCALL  CLEF$S
                                        .ENDC
000000                          QIOLBL: QIO$            ;JUST DEFINE QIO OFFSETS
        000000'                 .=QIOLBL                ;OVERLAY QIO AREA

000000                          GMCR:   GMCR$           ;GET MCR COMMAND LINE
                                        .ENDC
                                        .IF DF RT11
```

```
INITAL  MACRO M1104  06-DEC-77 08:24  PAGE 4-1
COMTEX MACRO DEFINITIONS

                                .MCALL  ..V2..,.PRINT,.HRESET,.EXIT
                                .MCALL  .QSET,.RCTRLO
                                ..V2..
                                .ENDC
                            ; MACRO DEFINITIONS

                                .IF DF  RSX11M
                                .MACRO  DIV     DISOR,DEND
                                .IF DIF <DEND>,<R0>
                                MOV     DEND,R0
                                .IFF
                                MOV     R1,R0
                                .ENDC
                                .IIF DIF DISOR,R1       MOV     DISOR,R1
                                CALL    $DIV
                                .ENDM
                            ;
                                .MACRO  MUL     MIER,MCAND
                                .IIF DIF MIER,R0        MOV     MIER,R0
                                .IIF DIF MCAND,R1       MOV     MCAND,R1
                                CALL    $MUL
                                 ENDM
                                .ENDC
                            ;
```

```
INITAL  MACRO M1104  06-DEC-77 08:24  PAGE 5
INITALIZE LINE TABLES


                                        .SBTTL  INITALIZE LINE TABLES

                                ;+
                                ; **-INITAL - INITIALIZE LINE TABLES, DEVICE VECTORS ETC.
                                ;
                                ; INPUTS:
                                ;       NONE
                                ; OUTPUTS:
                                ;       EXITS WITH BUFFERS ASSIGNED AND LINE TABLES INITIALIZED
                                .IF DF RSX
                                ;       THE BISYNC TAP TIMER ARE ALSO CALCULATED BY THIS ROUTINE
                                ;       FOR THE RSX 2780'S. THE CRB TIMER AND THE SLAVE RECEIVE TIMER
                                ;       ARE SPEED AND CLOCK TICS PE SECOND DEPENDENT.  IN THE
                                ;       CALCULATIONS BELOW THE FORMULAS ARE BASED ON FOR A
                                ;       2000 BAUD LINE WITH A CLOCK RUNNING AT 60 TICS PER SECOND
                                ;       IF THESE ARE THE PARAMETERS OF THE TARGET SYSTEM THEN THE
                                ;       SLAVE RECEIVE TIMER WILL BE 102.  TICS AND THE CRB TIMER WILL
                                ;       BE 5.  TICS
                                .ENDC
                                ;
                                ; CALLING SEQUENCE:
                                ;       JSR     PC,INITAL
                                ;-
                                        .GLOBL  INITAL,DONE,RCVBLK,TRNSBL
                                        .GLOBL  BASELN,COMAV,CMBFAV
                                        .IF DF RT11
                                        .GLOBL  QLSTRT
                                        .ENDC
                                ;
                                        .ENABL  LSB
000122  005066  000002          INITAL: CLR     2(SP)           ;RESET RCVE BCC ERROR COUNT
000126  112767  000000G 000000G        MOVB    #COMAV,CMBFAV           ;PRESET COMM BUFFERS AVAIL
                                        .IF DF  RT11
                                        .RCTRLO                         ;RESET CONTROL O
                                        .QSET   #QLSTRT,#10.    ;SET UP FOR EXTRA QUEUE ELEMENTS
                                        BISB    #100,@#44       ;SET UP FOR NO STALL ON TTY INPUT
                                                                ;IMPORTANT IN F/B
                                        .ENDC
000134  012767  000777  000330        MOV     #RSTRT,INTPAR           ;SELECT RESTART LINIT
000142                                LINIT   INTPAR                  ;DO RESTART LINIT
000152  100503                        BMI     10$                     ;CAN'T RESTART
000154  012767  000000G 000310        MOV     #BASELN,INTPAR          ;SELECT START
                                        .IF DF  RSX
000162  012703  000472'               MOV     #INTPAR,R3              ;POINTER TO PARAMETER LIST
000166  062703  000006                ADD     #6,R3                   ;POINT TO TIMER AREA
000172  012701  000003                MOV     #3,R1                   ;MOVE CONSTANT TO R1
000176                                MUL     #D.SPED,R1              ;MULTIPLY BY LINE SPEED
000206  010146                        MOV     R1,-(SP)                ;SAVE DIVISOR ON STACK
                                        .IF DF  RSX11M
000210  016701  000000G               MOV     $TKPS,R1                ;GET TICKS PER SECOND
                                        .IFF
                                        MOV     .TKPS,R1                ;GET TICKS PER SECOND
                                        .ENDC
000214  010123                        MOV     R1,(R3)+                ;SAVE AS BID TIMER
000216  006201                        ASR     R1                      ;# OF TICKS IN 1/2 SECOND
000220  010123                        MOV     R1,(R3)+                ;SAVE AS INTERVAL TIMER TO
                                                                        ;CHECK SLAVE AND MASTER TIMEOUTS
```

```
INITAL  MACRO M1104  06-DEC-77 08:24  PAGE 5-1
INITALIZE LINE TABLES

    140 000222  012723  000006                  MOV     #6,(R3)+                ;MASTER TIMEOUT VALUE IS 6, 1/2
    141                                                                         ; SECOND INTERVALS
    142 000226  012723  000006                  MOV     #6,(R3)+                ;SAVE SLAVE RECEIVE TIMEOUT VALUE
    143 000232  005000                          CLR     R0                      ;CLEAR WORKING REGISTER
    144                                         .IF DF  RSX11M
    145 000234  016701  000000G                 MOV     $TKPS,R1                ;GET TICKS PER SECOND
    146                                         .IFF
    147                                         MOV     .TKPS,R1                ;GET TICKS PER SECOND
    148                                         .ENDC
    149 000240                                  MUL     #5,R1                   ;MULTIPLY BY 5
    150 000250                                  DIV     (SP)+,R0                ;DIVIDE BY CONSTANT
    151 000260  010013                          MOV     R0,(R3)                 ;SAVE CRB TIMER
    152                                         .ENDC
    153 000262                                  LINIT   INTPAR                  ;INITIALIZE THE LINE
    154 000272  100433                          BMI     10$                     ;NO GOOD
    155                                         .IF DF LSI11
    156                                         MTPS    #0              ;CLEAR PROCESSOR STATUS WORD
    157                                         .IFF
    158 000274  105037  177776                  CLRB    @#PS                    ;CLEAR PROCESSOR STATUS WORD
    159                                         .ENDC
    160 000300  005077  000000G                 CLR     @TRNSBL         ;NO TRNSMT FILES YET
    161                                         .IF DF  RSX
    162 000304  012777  177777  000000G         MOV     #-1,@RCVBLK     ;OR RECEIVE FILES
    163                                         .ENDC
    164                                         .IF DF  RT11
    165                                         CLR     @RCVBLK         ;OR RECEIVE FILES
    166                                         .PRINT  #OUMES
    167                                         RTS     PC              ;AND RETURN
    168                                 OUMES:  .ASCIZ  "RT11 RBT V03-00"
    169                                         .ENDC
    170                                         .IF DF  RSX
    171 000312                                  DIR$    #GMCR           ;THIS DIRECTIVE IS DONE MERELY TO
    172                                                                 ;FREE THE MCR BUFFER SO THAT ANOTHER
    173                                                                 ;TASK CAN USE IT.  NOTHING IS DONE
    174                                                                 ;WITH THE DATA READ.
    175 000320  012767  000011' 000014G         MOV     #HELMSG,QDPBM+Q.IOPL    ;ADDR OF MESSAGE
    176 000326  012767  000012  000016G         MOV     #HELLNG,QDPBM+Q.IOPL+2  ;LENGTH IN BYTES
    177 000334                                  DIR$    #QDPBM          ;QUEUE THE WRITE
    178 000342                                  WTSE$S  #5              ;WAIT FOR ITS COMPLETION
    179                                         .IF NDF RSX11M
    180                                         CLEF$S  #5
    181                                         .ENDC
    182 000354                                  FINIT$                          ;INITIALIZE FCS
    183 000360  000207                          RTS     PC                      ;AND RETURN
    184                                          ENDC
    185                                 ;
    186 000362                          10$:    ERRORK  <LINE INIT>
    187 000372  000167  000054                  JMP     DONE1
    188                                         .DSABL  LSB
    189                                 ;
    190
    191                                         .IF DF  RSX
    192 000011                                  .PSECT  MESAG2
    193 000011      122     112     105 HELMSG: .ASCII  "RJE V03-00"
        000014      040     126     060
        000017      063     055     060
        000022      060
```

INITALIZE LINE TABLES

```
    194          000012                    HELLNG=. -HELMSG
    195 000376                                     PSECT  MAIN
    196                                           .ENDC
```

```
                                        ;+
                                        ; **-DONE - DE-INITIALIZE AND EXIT ROUTINE
                                        ;
                                        ; INPUTS:
                                        ;       NONE
                                        ; OUTPUTS:
                                        ;       EXITS
                                        ; CALLING SEQUENCE:
                                                .IF DF RT11
                                        ;       JMP     DONE
                                                .ENDC
                                                .IF DF RSX
                                        ;       JSR     PC,DONE
                                                .ENDC
                                        ;-
                                                .IF DF RSX
000376                                  DONE::  WTSE$S  #5              ;WAIT FOR MESSAGE'S COMPLETION
000410  012700  000000G                         MOV     #TABSST,R0      ;CLEAR OUT SST VEC SO DON'T LOOP ON ERROR
000414  012705  000007                         MOV     #7,R5           ;
000420  005020                          5$:     CLR     (R0)+   ;
000422  005305                                 DEC     R5              ;
000424  001375                                 BNE     5$              ;
000426  012767  000777  000036                 MOV     #RSTRT,INTPAR   ;SET UP FOR RESTART
000434                                         LINIT   INTPAR          ;AND RESTART (DISCONNECT FROM VECTORS)
000444                                         EXIT$S
000452                                  DONE1:
000452                                         WTSE$S  #5              ;WAIT FOR MESSAGE'S COMPLETION
000464                                         EXIT$S                  ;EXIT - DON'T DO RESTART IN CASE SOMEONE
                                                                        ;ELSE IS USING CSR
                                                .ENDC

                                                .IF DF RT11
                                        DONE:   MOV     #RSTRT,INTPAR   ;SELECT RESTART
                                                MOV     #INTPAR,-(SP)   ;SET PARAMETER BLOCK ADDR ON STACK
                                                LINIT                   ;AND RESTART (DISCONNECT FROM VECTORS)
                                                .HRESET                 ;KILL I/O; RESET MONITOR
                                                CLR     R0
                                                .EXIT                   ;RETURN TO RT-11 MONITOR
                                                .ENDC
```

```
INITAL  MACRO M1104  06-DEC-77 08:24  PAGE 7
INITALIZE LINE TABLES


                                        ;+
                                        ; PARAMETER LIST
                                        ;-
                 000777                         RSTRT=777              ;RESTART LINIT PARAMETER
                                                                       ;LINE 377, OPCODE 1
                 000003                         DPISR=3                ;DP11 ISR CODE
                 000001                         BITAP=1                ;BISYNC TAP CODE
                                        ;
000472      000G                        INTPAR: .BYTE   BASELN         ;LINE NUMBER
000473      000                             .BYTE   0                  ;OP CODE
000474      003         001                 .BYTE   DPISR,BITAP        ;DP11 ISR AND BISYNC TAP
000476      000000                          .WORD   0                  ;FOR ERROR CODE
                                            .IF DF  RT11
                                            .WORD   CR.TIC             ;TICK COUNT FOR BISYNC TAP
                                            .ENDC
                                            .IF DF  RSX
000500      000000                          .WORD   0                  ;CONTENTION TIMER (TICKS)
000502      000000                          .WORD   0                  ;INTERVAL FOR MASTER/SLAVE TIMERS (TICKS)
000504      000000                          .WORD   0                  ;MASTER TIMER (INTERVALS)
000506      000000                          .WORD   0                  ;SLAVE TIMER (INTERVALS)
000510      000000                          .WORD   0                  ;CRB TIMER (TICKS)
                                            .ENDC
                                        ;
                 000001                         .END
```

```
INITAL  MACRO M1104  06-DEC-77 08:24  PAGE 7-1
SYMBOL TABLE

BASELN= ****** G        GMCR    000000R     002 QDPBM = ****** GX       RCVBLK= ****** G        TRNSBL= ****** G
BITAP = 000001          G.MCRB= 000002          Q1OLBL  000000R     002 RSTRT = 000777          TYPMSG= ****** GX
CMBFAV= ****** G        HELLNG= 000012          Q.IOAE= 000012          RSX   = 000001          $DIV  = ****** GX
COMAV = ****** G        HELMSG  000011R     003 Q.IOEF= 000006          RSX11M= 000001          $MUL  = ****** GX
DONE    000376RG    002 INITAL  000122RG    002 Q.IOFN= 000002          TABSST= ****** GX       $TKPS = ****** GX
DONE1   000452R     002 INTPAR  000472R     002 Q.IOLU= 000004          TAGE1   000011R     003 $$$ARG= 000000
DPISR = 000003          NRCVSF= 000000          Q.IOPL= 000014          TAGNUM= 000001          $$$OST= 000122
D.SPED= ****** GX       NTRNSF= 000002          Q.IOPR= 000007          TAG1    000000R     003 .FINIT= ****** G
EMTINT= ****** GX       PS    = 177776          Q.IOSB= 000010

.  ABS.   000000    000
          000000    001
MAIN      000512    002
MESAG2    000023    003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:   1832 WORDS  ( 8 PAGES)
DYNAMIC MEMORY:   2810 WORDS  ( 10 PAGES)
ELAPSED TIME:   00:00:26
 256,020 INITAL:27, 256,030 INITAL/CRF/-SP=UAPRXM,CTXMAC,INITAL
```

INITAL CREATED BY MACRO ON 6-DEC-77 AT 08:24 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES							
BASELN	= ******	G	5-107	5-124	7-246					
BITAP	= 000001		#7-244	7-248						
CMBFAV	= ******	G	5-107	*5-114						
COMAV	= ******	G	5-107	5-114						
DONE	000376	RG	5-106	#6-214						
DONE1	000452	R	5-187	#6-223						
DPISR	= 000003		#7-243	7-248						
D.SPED	= ******	GX	5-129							
EMTINT	= ******	GX	5-122	5-153	6-221					
GMCR	000000	R	#4-55	5-171						
G.MCRB	= 000002		4-55							
HELLNG	= 000012		5-176	#5-194						
HELMSG	000011	R	5-175	#5-193	5-194					
INITAL	000122	RG	5-106	#5-113						
INTPAR	000472	R	*5-121	5-122	*5-124	5-126	5-153	*6-220	6-221	#7-246
LSI11	= ******		5-155							
NRCVSF	= 000000		#1-5							
NTRNSF	= 000002		#1-4							
PS	= 177776		#4-44	5-158						
QDPBM	= ******	GX	*5-175	*5-176	5-177					
QIOLBL	000000	R	#4-52	4-53						
Q.IOFN	= 000002		4-52							
Q.IOPL	= 000014		*5-175	*5-176						
RCVBLK	= ******	G	5-106	5-162						
RSTRT	= 000777		5-121	6-220	#7-241					
RSX	= 000001		#1-2	4-37	4-47	5-92	5-125	5-161	5-170	5-191
			6-213	7-253						6-209
RSX11M	= 000001		#1-3	4-49	4-64	5-131	5-144	5-179		
RT11	= ******		4-40	4-57	5-108	5-115	5-164	6-206	6-229	7-250
TABSST	= ******	GX	6-215							
TAGE1	000011	R	5-186	#5-186						
TAGNUM	= 000001		#1-8	5-186	#5-186	5-186				
TAG1	000000	R	5-186	5-186	#5-186					
TRNSBL	= ******	G	5-106	5-160						
TYPMSG	= ******	GX	5-186							
$DIV	= ******	GX	5-150							
$MUL	= ******	GX	5-129	5-149						
$TKPS	= ******	GX	5-132	5-145						
$$$ARG	= 000000		#4-52	4-52	4-52					
$$$GLB	= ******		4-52	4-55						
.FINIT	= ******	G	5-182	5-182						

MACRO NAME	REFERENCES									
ASRBUF	#3-28									
CALL	#5-129	#5-149	#5-150	#6-222	#6-225					
DIR$	#4-48	5-171	5-177	#5-178	5-178	#6-214	6-214	#6-222	6-222	#6-224
	6-224	#6-225	6-225							
DIV	#4-65	#5-150								
ERROR	#1-9									
ERRORK	#1-16	5-186								
ERRORM	#1-32									
ERRORP	#1-47									
EXIT$S	#4-48	#6-222	#6-225							
FATAL	#1-53									
FINIT$	#4-48	#5-182								
GET$	#3-43									
GMCR$	#4-48	#4-55								
LIMIT	#3-23	5-122	5-153	6-221						
MOV$	#5-178	5-178	#6-214	6-214	#6-224	6-224				
MUL	#4-75	#5-129	#5-149							
OFF$	#4-52	#4-55								
PUTD	#3-52									
PUTMC	#3-38									
PUTTC	#3-33									
PUTTM	#3-47									
QDPB$	#4-52	#4-52								
QIO$	#4-48	4-52								
WTSE$S	#4-48	#5-178	#6-214	#6-224						

```
                        ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                  RSX=1
000001                  RSX11M=1
000002                  NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                  NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                        ;
                        ;MACRO DEFINITIONS
000000                          TAGNUM=0
                                .MACRO  ERROR   TEXT
                                ERRORK  <TEXT>
                                JSR     PC,PROMPT
                                .ENDM
                        ;
                        ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                        ;
                                .MACRO  ERRORK  TEXT
                                TAGNUM=TAGNUM+1
                                .IRP    N,<\TAGNUM>
                                JSR     R5,TYPMSG
                                .WORD   TAG'N
                                .WORD   TAGE'N-TAG'N
                                .PSECT  MESAG2,GBL
                        TAG'N:  .ASCII  "TEXT"
                        TAGE'N:
                                .PSECT  MAIN
                                .ENDM
                                .ENDM

                        ; MACRO TO QUEUE A MESSAGE ONLY
                        ;
                                .MACRO  ERRORM  TEXT
                                TAGNUM=TAGNUM+1
                                .IRP    N,<\TAGNUM>
                                JSR     R5,TYPMS1
                                .WORD   TAG'N
                                .WORD   TAGE'N-TAG'N
                                .PSECT  MESAG2,GBL
                        TAG'N:  .ASCII  "TEXT"
                        TAGE'N:
                                .PSECT  MAIN
                                .ENDM
                                .ENDM

                        ; MACRO TO QUEUE A MESSAGE AND PROMPT
                        ;
                                .MACRO  ERRORP  TEXT
                                ERRORM  <TEXT>
                                JSR     PC,PROMPT
                                .ENDM

                        ;
                                .MACRO  FATAL   TEXT
                                ERRORK  <TEXT>
                                JSR     R5,DIE
                                .ENDM
```

```
                                        .TITLE  NXTFIL
                                        .IDENT  /00006/
                                ;
                                ; COPYRIGHT (C)  1975, 1976, 1977
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                                ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                                ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                                ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                                ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                                ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                                ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                                ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                                ;
                                ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                                ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                                ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                                ;
                                ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                                ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                                ; DEC.
                                ;
                                ; OPEN FOR READ NEXT FILE IN COMMAND STRING
                                ; LAST MODIFICATION ON 16-JANUARY-75
                                ;
                                ;
000000                                  .PSECT  MAIN
                                ;
                                ; MACRO CALLS
                                        .MCALL  GET$S,CLOSE$,OFNB$P,FDOFF$
000000                                  FDOFF$  DEF$L           ;DEFINE FCS OFFSETS LOCALLY
                                ;
                                ;+
                                ; INPUTS:
                                ;       TRNSBLK = POINTER TO LAST FILE NAME OPENNED
                                ;
                                ; CALL:
                                ;       NXTFIL - TO OPEN NEXT FILE
                                ;       FSTFIL - TO OPEN FIRST FILE IN A COMMAND STRING (TRANSMIT)
                                ;
                                ; OUTPUTS:
                                ;       ROUTINE UPDATES TRNSBLK AND OPENS NEXT FILE IF ANY
                                ;       RETURNS WITH "C" BIT SET IF AN OPEN FAILURE OCCURRED
                                ;       RETURNS WITH "C" AND "Z" BIT SET IF NO MORE FILES FROM
                                ;       COMMAND STRING
                                ;-
                                        .IF DF  RSX
000000                          NXTFIL::CLOSE$  #FDBIN          ;FIRST CLOSE THE FILE
000010  062767  000036G 000000G         ADD     #S.FNB+XMTSWT,TRNSBLK   ;GET LOCATION OF NEXT FNB
000016  016700  000000G         FSTFIL::MOV     TRNSBLK,R0      ;
000022  005710                          TST     (R0)
000024  001427                          BEQ     30$             ;LAST FNB FOR THIS COMMAND
000026  012701  000017                  MOV     #S.FNBW,R1      ;NUMBER OF WORDS IN FNB
000032  012705  000102G                 MOV     #FDBIN+F.FNB,R5 ;FDB FOR INPUT
000036  012025                  10$:    MOV     (R0)+,(R5)+     ;MOVE THE FNB
000040  005301                          DEC     R1
```

```
 58 000042  001375                        BNE     10$                 ;NOT FINISHED
 59 000044  012701  000000C               MOV     #XMTSWT/2,R1        ;# OF WORDS OF XMIT SWITCHES
 60 000050  012705  000000G               MOV     #SWTBN,R5           ;ADDR OF FIRST SWITCH
 61 000054  012025                 15$:    MOV     (R0)+,(R5)+         ;MOVE A SWITCH VALUE
 62 000056  005301                         DEC     R1                  ;ONE LESS SWITCH
 63 000060  001375                         BNE     15$                 ;MOVE THE NEXT ONE
 64 000062                                 OFNB$R  #FDBIN              ;OPEN
 65 000100  000244                 20$:    CLZ                 ;ANOTHER FILE AT LEAST WAS FOUND
 66 000102  000401                         BR      40$
 67 000104  000261                 30$:    SEC                         ;INDICATE ERROR CONDITION EXISTS
 68                                                                    ;NO FILES ARE OPENNED
 69 000106  000207                 40$:    RTS     PC
```

```
                                ;+
                                ; **-FSTGET - ROUTINE TO DO THE FIRST GET AFTER THE FILE IS OPENNED
                                ;             AND POINTED TO IF NECESSARY
                                ;
                                ; INPUTS:
                                ;       NONE
                                ;
                                ; OUTPUTS:
                                ;
                                ;       SEQUENTIAL GET ON THE FILE OPENNED IN NXTFIL IS DONE
                                ;       R0 IS DESTROYED.
                                ;-
                                        .ENABL  LSB
000110                          FSTGET.:GET$S   #FDBIN,,        ;DO THE GET
000120  103013                          BCC     30$             ;NO ERRORS
000122  122760  000006 000052           CMPB    #IE.EOF,F.ERR(R0)       ;END OF FILE
000130  001406                          BEQ     20$             ;YES, NO ERROR MESSAGE
                                                                ; THIS CAN OCCURR IF AN RVI WAS RECEIVED
                                                                ; ON THE NEXT TO LAST RECORD. WHEN THE
                                                                ; LAST RECORD IS READ THE GET IS NOT DONE
                                                                ; AND HENCE THE EOF CONDITION
                                                                ; IS NOT DETECTED
000132                                  ERROR   <IN ERR>        ;GIVE ERROR MESSAGE
000146  000261                  20$:    SEC                     ;SET CARRY
000150  000207                  30$:    RTS     PC              ;RETURN
                                        .DSABL  LSB
                                        .ENDC
        000001                          .END
```

```
NXTFIL  MACRO M1104  06-DEC-77 08:25  PAGE 3-1
SYMBOL TABLE

FDBIN = ****** GX        F.DSPT= 000044          F.NREC= 000030          NTRNSF= 000002          S.FDB = 000140
FO.RD = ****** GX        F.DVNM= 000134          F.OVBS= 000030          NXTFIL  000000RG  002 S.FNAM= 000006
FSTFIL  000016RG    002  F.EFBK= 000010          F.RACC= 000016          N.DID = 000024          S.FNB = 000036
FSTGET  000110RG    002  F.EFN = 000050          F.RATT= 000001          N.DVNM= 000032          S.FNBW= 000017
F.ACTL= 000076           F.EOBB= 000032          F.RCNM= 000034          N.FID = 000000          S.FNTY= 000004
F.ALOC= 000040           F.ERR = 000052          F.RCTL= 000017          N.FNAM= 000006          S.FTYP= 000002
F.BBFS= 000062           F.FACC= 000043          F.RSIZ= 000002          N.FTYP= 000014          S.NFEN= 000020
F.BDB = 000070           F.FFBY= 000014          F.RTYP= 000000          N.FVER= 000016          TAGE1   000006R     003
F.BGBC= 000057           F.FNAM= 000110          F.SEQN= 000100          N.NEXT= 000022          TAGNUM= 000001
F.BKDN= 000026           F.FNB = 000102          F.SPDV= 000072          N.STAT= 000020          TAG1    000000R     003
F.BKDS= 000020           F.FTYP= 000116          F.SPUN= 000074          N.UNIT= 000034          TRNSBL= ****** GX
F.BKEF= 000050           F.FVER= 000120          F.STBK= 000036          PAR$$$= 000027          TYPMSG= ****** GX
F.BKP1= 000051           F.HIBK= 000004          F.UNIT= 000136          PROMPT= ****** GX       XMTSWT= ****** GX
F.BKST= 000024           F.LUN = 000042          F.URBD= 000020          RSX   = 000001          .CLOSE= ****** G
F.BKVB= 000064           F.MBCT= 000054          F.VBN = 000064          RSX11M= 000001          .GETSQ= ****** G
F.CHR = 000075           F.MBC1= 000055          F.VBSZ= 000060          SWTBN = ****** GX       .OPFNB= ****** G
F.CNTG= 000034           F.MBFG= 000056          IE.EOF= ****** GX       S.FATT= 000016          ...TPC= 000140
F.DFNB= 000046           F.NRBD= 000024          NRCVSF= 000000

.  ABS.   000000    000
          000000    001
MAIN     000152    002
MESAG2   000006    003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:   1933 WORDS  ( 8 PAGES)
DYNAMIC MEMORY:   2810 WORDS  ( 10 PAGES)
ELAPSED TIME:   00:00:15
 256,020 NXTFIL;27,  256,030 NXTFIL/CRF/-SP=UAPRXM,NXTFIL
```

NXTFIL CREATED BY MACRO ON 6-DEC-77 AT 08:25
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES						
FDBIN	= ******	GX	2-49	2-49	2-55	2-64	2-64	3-84	3-84
FO.RD	= ******	GX	2-64						
FSTFIL	000016	RG	#2-51						
FSTGET	000110	RG	#3-84						
F.ERR	= 000052		3-86						
F.FACC	= 000043		*2-64						
F.FNB	= 000102		2-55						
IE.EOF	= ******	GX	3-86						
NRCVSF	= 000000		#1-5						
NTRNSF	= 000002		#1-4						
NXTFIL	000000	RG	#2-49						
PARSSS	= 000027		#2-49	2-49	#2-64	2-64	#3-84	3-84	
PROMPT	= ******	GX	3-93						
RSX	= 000001		#1-2	2-48					
RSX11M	= 000001		#1-3						
SWTBN	= ******	GX	2-60						
S.FNB	= 000036		2-50						
S.FNBW	= 000017		2-54						
TAGE1	000006	R	3-93	#3-93					
TAGNUM	= 000001		#1-8	3-93	#3-93	3-93			
TAG1	000000	R	3-93	3-93	#3-93				
TRNSBL	= ******	GX	*2-50	2-51					
TYPMSG	= ******	GX	3-93						
XMTSWT	= ******	GX	2-50	2-59					
.CLOSE	= ******	G	2-49	2-49					
.GETSQ	= ******	G	3-84	3-84					
.OPFNB	= ******	G	2-64	2-64					

MACRO NAME	REFERENCES									
CLOSE$	#2-31	2-49								
CMOV$B	#2-64	#2-64	#2-64	#2-64	#2-64	#2-64				
CMOV$W	#2-64	#2-64	#2-64	#2-64	#2-64	#2-64	#2-64	#3-84	#3-84	#3-84
ERROR	#1-9	3-93								
ERRORK	#1-16	3-93								
ERRORM	#1-32									
ERRORP	#1-47									
ERR$	#2-49	#2-49	#2-64	#2-64	#3-84	#3-84				
FATAL	#1-53									
FDOFF$	#2-31	#2-32								
FDOP$R	#2-64	2-64								
FDRC$R	#2-64	#2-64								
GET$S	#2-31	#3-84								
LDFDB$	#2-49	2-49	#2-64	2-64	#2-64	2-64	#3-84	3-84		
OFNB$	#2-64	#2-64								
OFNB$P	#2-31	#2-64								

PARSE MACRO M1104 06-DEC-77 08:26 PAGE 1

```
                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM


                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                                        .TITLE  PARSE
                                        .IDENT  /00005/
                                ;
                                ; COPYRIGHT (C)  1975, 1076, 1077
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                                ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                                ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                                ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                                ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                                ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                                ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                                ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                                ;
                                ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                                ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                                ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                                ;
                                ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                                ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                                ; DEC.
                                ;
                                ; VERSION 0
                                ;
                                ; ROBERT H.  ROSENBAUM 1-AUGUST-74
                                ; LAST MODIFICATION ON 3-MARCH-75
                                ; CHANGED TO AVOID DEFAULT FILE SPEC DELETEING OTHER SWITCHES
                                ; CLARISE E.  PATTON 9/77
                                ;
                                ; PARSE OUTPUT FILES
                                ;
000000                                  .PSECT  MAIN
                                ;
                                ; MACRO CALLS
                                        .MCALL  CSI$1,CSI$2,CSI$SW,CSI$SV,CSI$ND,FDOFF$,FCSBT$,CALL
                                        .MCALL  DIR$
000000                                  FCSBT$  DEF$L
000000                                  FDOFF$  DEF$L
                                ;
                                ;+
                                ; INPUTS:
                                ;       TAKES ITS INPUT FROM THE TELETYPE COMMAND STRING
                                ;       R2 = RCVBLK+2 (BLOCK WHERE INPUT FILES ARE TO BE PARSED TO)
                                ; OUTPUTS:
                                ;       FILLS IN THE RECEIVE BLOCK (RCVBLK) AS THE INFO MUST
                                ;       APPEAR IN THE FDB FOR THE OPEN
                                ;-
                                        .IF DF  RSX
                                        .ENABL  LSB             ;ENABLE LOCAL SYMBOL BLOCK
000000                          PENTER::CSI$1   #FILDES,FILDES+C.CMLD+2,FILDES+C.CMLD
000024  103531                         BCS     70$
000026  010201                         MOV     R2,R1           ;RCVBLK AREA
000030                          10$:    CALL    CLRRSW          ;INITIALIZE COMMON RECV SWITCH AREA FOR PARSE
000034  005067  000000G                CLR     SWRDF           ;CLEAR DF SWITCH TOO
000040                                  CSI$2   #FILDES,OUTPUT,#SWRAB
000062  103501                         BCS     60$
```

PARSE MACRO M1104 06-DEC-77 08:26 PAGE 2-1

```
 58 000064  012700  000000G                    MOV     #FDBOUO,R0      ;FDB FOR OUPUT FILES
 59 000070  012702  000006G                    MOV     #FILDES+C.DSDS,R2       ;DATASET POINTER
 60 000074  012703  000000G                    MOV     #DFBLK,R3       ;DEFAULT FILE NAME BLOCK
 61 000100  010146                             MOV     R1,-(SP)        ;SAVE LOCATION TO PARSE TO
 62 000102  105767  000000G                    TSTB    SWRDF           ;REQUEST TO CHANGE DEFAULT RECV FILE?
 63 000106  001402                             BEQ     12$             ;NO
 64 000110  012701  000000G                    MOV     #DFNLP,R1       ;YES, PARSE TO DEFAULT NAME BLOCK
 65 000114                          12$:       CALL    .PARSE
 66 000120  012601                             MOV     (SP)+,R1        ;RESTORE R1 AND STACK POINTER
 67 000122  103452                             BCS     50$             ;ERROR FOUND FROM PARSE
 68 000124  132761  000070  000020             BITB    #NB.SVR!NB.SNM!NB.STP,N.STAT(R1)
 69 000132  001066                             BNE     70$             ;BAD FILE OR VERSION
 70 000134  105767  000000G                    TSTB    SWRDF           ;REQUEST TO CHANGE DEFAULT RECV FILE?
 71 000140  001014                             BNE     17$             ;YES SKIP SWITCH PROCESSING
 72 000142  062701  000036                     ADD     #S.FNB,R1       ;NUMBER OF BYTES IN FNB
 73 000146  012704  000000G                    MOV     #SWRBN,R4       ;ADDRESS OF PERMANENT SWITCHES
 74 000152  012700  000000C                    MOV     #RCVSW1/2,R0    ;# OF WORDS OF RECEIVE SWITCHES
 75 000156  012703  000000G                    MOV     #SWRB1,R3       ;ADDR OF FIRST TEMPORARY SWITCH
 76 000162  011324                  15$:       MOV     (R3),(R4)+      ;MOVE TO PERMANENT SWITCHES
 77 000164  012321                             MOV     (R3)+,(R1)+     ;MOVE TO RECEIVE BLOCK AREA
 78 000166  005300                             DEC     R0              ;ONE LESS SWITCH
 79 000170  001374                             BNE     15$             ;MOVE THE NEXT ONE
 80 000172  132767  000020  000001G 17$:       BITB    #CS.MOR,C.STAT+FILDES
 81 000200  001415                             BEQ     20$
 82 000202  020127  000000G                    CMP     R1,#RCVSE
 83 000206  103710                             BLO     10$             ;STILL MORE ROOM
 84 000210  012777  177777  000000G            MOV     #-1,@RCVBLK     ;TOO MANY FILES SPECIFIED
 85 000216                                     ERRORP  <TOO MANY RECV FILES>
 86 000232  000436                             BR      80$             ;TAKE ERROR EXIT
 87 000234  105767  000000G             20$:   TSTB    SWRDF           ;DID WE JUST PROCESS A DEFAULT FILE?
 88 000240  001002                             BNE     21$             ;YES
 89 000242  012711  177777                     MOV     #-1,(R1)        ;MARK END OF RECV BLOCK AREA
 90 000246  000207                  21$:       RTS     PC
 91 000250                          50$:       ERRORP  <PARSE FAIL>
 92 000264  000421                             BR      80$
 93 000266  004767  000000G             60$:   JSR     PC,CSIERR       ;PRINT THE STRING IN ERROR
 94 000272                                     ERRORP  <ILL SWITCH>
 95 000306  000410                             BR      80$
 96 000310  004767  000000G             70$:   JSR     PC,CSIERR       ;PRINT THE STRING IN ERROR
 97 000314                                     ERRORP  <WILD CARD FND>
 98 000330  012777  177777  000000G     80$:   MOV     #-1,@RCVBLK
 99 000336  000261                             SEC                     ;INDICATE ERROR EXISTS
100 000340  000207                             RTS     PC              ;RETURN TO CALLER
101                                            .DSABL  LSB
```

```
                                ;+
                                ;COMMAND STRING INTERPRETER'S SWITCH TABLE
                                ;-
000342                          SWRAB:
000342                                  CSI$SW  B1,200,SWRB1
000352                                  CSI$SW  SL,200,SWRS1
000362                                  CSI$SW  RC,200.RCHN1
000372                                  CSI$SW  FX,R.FIX,RTYP1,,,VALU
000402                                  CSI$SW  FN,FD.FTN,RATT1
000412                                  CSI$SW  TK,220,SWRB1
000422                                  CSI$SW  DF,200,SWRDF
000432                                  CSI$ND
000434                          VALU:   CSI$SV  NUMERIC,RVAL1,2
000440                                  CSI$ND
                                        .ENDC
        000001                          .END
```

```
PARSE   MACRO M1104  06-DEC-77 08:26  PAGE 3-1
SYMBOL TABLE

CHR$$ = 000106          FA.TMP= 000020          F.ALOC= 000040          F.RTYP= 000000          RSX    = 000001
CH.AND= 000001          FA.WCK= 020000          F.BBFS= 000062          F.SEQN= 000100          RSX11M= 000001
CLRRSW= ****** GX       FA.WRT= 000002          F.BDB = 000070          F.SPDV= 000072          RTYP1 = ****** GX
CNT$$ = 000002          FDBOUO= ****** GX       F.BGBC= 000057          F.SPUN= 000074          RVAL1 = ****** GX
CSIERR= ****** GX       FD.BLK= 000010          F.BKDN= 000026          F.STBK= 000036          R.FIX = 000001
CS.DIF= 000002          FD.CCL= 000002          F.BKDS= 000020          F.UNIT= 000136          R.SEQ = 000003
CS.DVF= 000004          FD.COM= 020000          F.BKEF= 000050          F.URBD= 000020          R.VAR = 000002
CS.EQU= 000040          FD.CR = 000002          F.BKP1= 000051          F.VBN = 000064          SIZ$$ = 000002
CS.INP= 000001          FD.DIR= 000010          F.BKST= 000024          F.VBSZ= 000060          SWRAB   000342R     002
CS.MOR= 000020          FD.FTN= 000001          F.BKVB= 000064          NB.DEV= 000200          SWRBN = ****** GX
CS.NMF= 000001          FD.F11= 040000          F.CHR = 000075          NB.DIR= 000100          SWRB1 = ****** GX
CS.OUT= 000002          FD.INS= 000010          F.CNTG= 000034          NB.NAM= 000004          SWRDF = ****** GX
CS.WLD= 000010          FD.ISP= 002000          F.DFNB= 000046          NB.SD1= 000400          SWRS1 = ****** GX
C.CMLD= 000002          FD.MNT= 100000          F.DSPT= 000044          NB.SD2= 001000          S.FATT= 000016
C.DEVD= 000006          FD.OSP= 004000          F.DVNM= 000134          NB.SNM= 000040          S.FDB = 000140
C.DIRD= 000012          FD.PLC= 000004          F.EFBK= 000010          NB.STP= 000020          S.FNAM= 000006
C.DSDS= 000006          FD.PSE= 010000          F.EFN = 000050          NB.SVR= 000010          S.FNB = 000036
C.FILD= 000016          FD.RAH= 000001          F.EOBB= 000032          NB.TYP= 000002          S.FNBW= 000017
C.MKW1= 000024          FD.RAN= 000002          F.ERR = 000052          NB.VER= 000001          S.FNTY= 000004
C.MKW2= 000026          FD.REC= 000001          F.FACC= 000043          NRCVSF= 000000          S.FTYP= 000002
C.SIZE= 000054          FD.RWM= 000001          F.FFBY= 000014          NTRNSF= 000002          S.NFEN= 000020
C.STAT= 000001          FD.SDI= 000020          F.FNAM= 000110          N.DID = 000024          TAGE1   000023R     003
C.SWAD= 000022          FD.SQD= 000040          F.FNB = 000102          N.DVNM= 000032          TAGE2   000035R     003
C.TYPR= 000000          FD.TTY= 000004          F.FTYP= 000116          N.FID = 000000          TAGE3   000047R     003
DFBLK = ****** GX       FD.WBH= 000002          F.FVER= 000120          N.FNAM= 000006          TAGE4   000064R     003
DFNLP = ****** GX       FF.CHR= 000005          F.HIBK= 000004          N.FTYP= 000014          TAGNUM= 000004
FA.APD= 000100          FF.NV = 000003          F.LUN = 000042          N.FVER= 000016          TAG1    000000R     003
FA.CRE= 000010          FF.POE= 000002          F.MBCT= 000054          N.NEXT= 000022          TAG2    000023R     003
FA.DLK= 001000          FF.RWD= 000001          F.MBC1= 000055          N.STAT= 000020          TAG3    000035R     003
FA.ENB= 100000          FF.RWF= 000006          F.MBFG= 000056          N.UNIT= 000034          TAG4    000047R     003
FA.EXC= 002000          FF.SPC= 000004          F.NRBD= 000024          PAR$$$= 000027          TYPMS1= ****** GX
FA.EXT= 000004          FILDES= ****** GX       F.NREC= 000030          PENTER  000000RG    002 VALU    000434R     002
FA.NSP= 000100          FO.APD= 000106          F.OVBS= 000030          PROMPT= ****** GX       $$     = 000000
FA.POS= 010000          FO.MFY= 000002          F.RACC= 000016          RATT1 = ****** GX       .CSI1 = ****** G
FA.RD = 000001          FO.RD = 000001          F.RATT= 000001          RCHN1 = ****** GX       .CSI2 = ****** G
FA.RWD= 004000          FO.UPD= 000006          F.RCNM= 000034          RCVBLK= ****** GX       .PARSE= ****** GX
FA.SEQ= 040000          FO.WRT= 000016          F.RCTL= 000017          RCVSE = ****** GX       ...GBL= 000000
FA.SHR= 000040          F.ACTL= 000076          F.RSIZ= 000002          RCVSW1= ****** GX       ...TPC= 000140

.  ABS.    000000      000
           000000      001
MAIN       000442      002
MESAG2     000064      003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:   3940 WORDS  ( 16 PAGES)
DYNAMIC MEMORY:   4922 WORDS  ( 18 PAGES)
ELAPSED TIME:   00:00:33
 256,020 PARSE;27, 256,030 PARSE/CRF/-SP=UAPRXM,PARSE
```

PARSE CREATED BY MACRO ON 6-DEC-77 AT 08:26 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES								
CHR$$	= 000106		#3-107	3-107	3-107	#3-107	3-107	3-107	#3-108	3-108	3-108
			#3-108	3-108	3-108	#3-109	3-109	3-109	#3-109	3-109	3-109
			#3-110	3-110	3-110	#3-110	3-110	3-110	#3-111	3-111	3-111
			#3-111	3-111	3-111	#3-112	3-112	3-112	#3-112	3-112	3-112
			#3-113	3-113	3-113	#3-113	3-113	3-113			
CLRRSW	= ******	GX	2-54								
CNT$$	= 000002		#3-107	3-107	#3-107	3-107	3-107	#3-107	3-107	#3-108	3-108
			#3-108	3-108	3-108	#3-108	3-108	#3-109	3-109	#3-109	3-109
			3-109	#3-109	3-109	#3-110	3-110	#3-110	3-110	3-110	#3-110
			3-110	#3-111	3-111	#3-111	3-111	3-111	#3-111	3-111	#3-112
			3-112	#3-112	3-112	3-112	#3-112	3-112	#3-113	3-113	#3-113
			3-113	3-113	#3-113	3-113					
CSIERR	= ******	GX	2-93	2-96							
CS.MOR	= 000020		2-80								
CS.OUT	= 000002		2-56								
C.CMLD	= 000002		2-51	*2-51	2-51	*2-51					
C.DSDS	= 000006		2-59								
C.SIZE	= 000054		2-51	2-56							
C.STAT	= 000001		2-80								
C.SWAD	= 000022		*2-56								
DFBLK	= ******	GX	2-60								
DFNLP	= ******	GX	2-64								
FDBOUO	= ******	GX	2-58								
FD.FTN	= 000001		3-111								
FILDES	= ******	GX	2-51	2-51	2-51	2-51	2-56	2-56	2-59	2-80	
NB.SNM	= 000040		2-68								
NB.STP	= 000020		2-68								
NB.SVR	= 000010		2-68								
NRCVSF	= 000000		#1-5								
NTRNSF	= 000002		#1-4								
N.STAT	= 000020		2-68								
PAR$$$	= 000027		#2-51	2-51	#2-56	2-56					
PENTER	000000	RG	#2-51								
PROMPT	= ******	GX	2-85	2-91	2-94	2-97					
RATT1	= ******	GX	3-111								
RCHN1	= ******	GX	3-109								
RCVBLK	= ******	GX	2-84	2-98							
RCVSE	= ******	GX	2-82								
RCVSW1	= ******	GX	2-74								
RSX	= 000001		#1-2	2-49							
RSX11M	= 000001		#1-3								
RTYP1	= ******	GX	3-110								
RVAL1	= ******	GX	3-115								
R.FIX	= 000001		3-110								
SIZ$$	= 000002		#3-107	3-107	3-107	3-107	#3-108	3-108	3-108	3-108	#3-109
			3-109	3-109	3-109	#3-110	3-110	3-110	3-110	#3-111	3-111
			3-111	3-111	#3-112	3-112	3-112	3-112	#3-113	3-113	3-113
			3-113								
SWRAB	000342	R	2-56	#3-106							
SWRBN	= ******	GX	2-73								
SWRB1	= ******	GX	2-75	3-107	3-112						
SWRDF	= ******	GX	*2-55	2-62	2-70	2-87	3-113				

SYMBOL	VALUE		REFERENCES								
SWRS1	= ******	GX	3-108								
S.FNB	= 000036		2-72								
TAGE1	000023	R	2-85	#2-85							
TAGE2	000035	R	2-91	#2-91							
TAGE3	000047	R	2-94	#2-94							
TAGE4	000064	R	2-97	#2-97							
TAGNUM	= 000004		#1-8	*2-85	#2-85	2-85	2-91	#2-91	2-91	2-94	#2-94
			2-94	2-97	#2-97	2-97					
TAG1	000000	R	2-85	2-85	#2-85						
TAG2	000023	R	2-91	2-91	#2-91						
TAG3	000035	R	2-94	2-94	#2-94						
TAG4	000047	R	2-97	2-97	#2-97						
TYPMS1	= ******	GX	2-85	2-91	2-94	2-97					
VALU	000434	R	3-110	#3-115							
$$	= 000000		#3-107	3-107	#3-107	3-107	#3-107	3-107	#3-107	3-107	#3-108
			3-108	#3-108	3-108	#3-108	3-108	#3-108	3-108	#3-109	3-109
			#3-109	3-109	#3-109	3-109	#3-109	3-109	#3-110	3-110	#3-110
			3-110	#3-110	3-110	#3-110	3-110	#3-111	3-111	#3-111	3-111
			#3-111	3-111	#3-111	3-111	#3-112	3-112	#3-112	3-112	#3-112
			3-112	#3-112	3-112	#3-113	3-113	#3-113	3-113	#3-113	3-113
			#3-113	3-113							
.CS11	= ******	G	2-51	2-51							
.CS12	= ******	G	2-56	2-56							
.PARSE	= ******	GX	2-65								

MACRO NAME	REFERENCES							
CALL	#2-36	#2-54	#2-65					
CSI$	#2-51	#2-56						
CSI$ND	#2-36	#3-114	#3-116					
CSI$SV	#2-36	#3-115						
CSI$SW	#2-36	#3-107	#3-108	#3-109	#3-110	#3-111	#3-112	#3-113
CSI$1	#2-36	#2-51						
CSI$2	#2-36	2-56						
DIR$	#2-37							
ERROR	#1-9							
ERRORK	#1-16							
ERRORM	#1-32	#2-85	#2-91	#2-94	#2-97			
ERROPP	#1-47	2-85	2-91	2-94	2-97			
FATAL	#1-53							
FCSBT$	#2-36	2-38						
FDOFF$	#2-36	2-39						
_DPD$	#2-51	2-51	#2-56	2-56				

```
QUEHAN  MACRO M1104  06-DEC-77 08:26  PAGE 1


                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM

                                ;
                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                                .TITLE  QUEHAN
                                .IDENT  /00002/
                        ;
                        ; COPYRIGHT (C)  1975, 1976, 1977
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                        ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                        ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                        ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                        ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                        ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                        ;
                        ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ; DEC.
                        ;
                        ; VERSION 0
                        ;
                        ; ROBERT H. ROSENBAUM 1-AUGUST-74
                        ;
                        ; LAST MODIFICATION ON 20-NOVEMBER-74
                        ;
                        ; QUERY COMMAND HANDLER
                        ;
000000                          .PSECT  MAIN
                        ;
                        ; GLOBAL DEFINITIONS
                                .GLOBL  QDPBM,FDBIN,FDBOUO

                        ; MACRO CALLS
                                .MCALL  FDOFF$,DIR$,QIO$
000000                          FDOFF$  DEF$L
000000                  QIOLBL: QIO$            ;JUST DEFINE QIO OFFSETS
        000000'                 .=QIOLBL        ;OVERLAY QIO AREA
                        ;
                        ;+
                        ; INPUTS:
                        ;       NONE
                        ; OUTPUTS:
                        ;       ROUTINE WILL PRINT ON THE OPERATORS CONSOLE THE CURRENT
                        ;       STATUS OF EMULATOR INCLUDING OFFLINE/ONLINE CONDITION,
                        ;       TRANSMIT/RECEIVE CONDITION, AND FILE NAME WHICH IT IS
                        ;       CURRENTLY ACCESSING.
                        ;-

000000                  QUEHAN::.IF DF  RT11
                                MOV     #QEND-QSTAT,R1  ;LENGTH OF MESSAGE
                                MOV     #QSTAT,R5
                                JSR     PC,R50STR       ;PRINT RAD50 STRING
                                MOV     #10,R1          ;TYPE DEVICE AND FILE NAME
```

QUEHAN MACRO M1104 06-DEC-77 08:26 PAGE 2-1

```
                                                        CMPB    MODE,#2          ;TRNSMIT?
                                                        BNE     QUER             ;NO
                                                        MOV     TRNSBLK,R5       ;ADDRESS OF DEV AND FILE NAME
                                                        BR      QFIL             ;PRINT MESSAGE
                                              QUER:     CMPB    MODE,#1          ;RECV
                                                        BNE     GOGET2           ;NO
                                                        MOV     RCVBLK,R5        ;ADDRESS OF DEVICE AND FILE NAME
                                              QFIL:     JSR     PC,R50STR        ;PRINT RAD50 STRING
                                                        .ENDC
                                                        .IF DF  RSX
000000  012702  000000G                                 MOV     #KBIBUF,R2       ;FOR QUE STOREAGE
000004  010267  000014G                                 MOV     R2,QDPBM+Q.IOPL
000010  012705  000000G                                 MOV     #QSTAT,R5        ;FOR STATUS MESSAGE
000014  020527  000000G                       10$:      CMP     R5,#QEND         ;FINSHED?
000020  103004                                          BHIS    20$              ;YES
000022  012500                                          MOV     (R5)+,R0         ;RAD50 TO ASCII
000024  004767  000000G                                 JSR     PC,R50OUT
000030  000771                                          BR      10$
000032  012767  000000G 000016G             20$:       MOV     #QLNGTH,QDPBM+Q.IOPL+2  ;NUMBER OF CHARACTERS TO PRINT
000040  012705  000102G                                 MOV     #FDBIN+F.FNB,R5
000044  126727  000000G 000002                          CMPB    MODE,#2          ;TRANSMIT MODE?
000052  001406                                          BEQ     30$              ;YES
000054  126727  000000G 000001                          CMPB    MODE,#1          ;RECEIVE MODE
000062  001051                                          BNE     40$              ;NO
000064  012705  000102G                                 MOV     #FDBOUO+F.FNB,R5
000070  116522  000032                       30$:       MOVB    N.DVNM(R5),(R2)+
000074  001444                                          BEQ     40$
000076  116522  000033                                 MOVB    N.DVNM+1(R5),(R2)+
000102  116503  000034                                 MOVB    N.UNIT(R5),R3    ;GET UNIT NUMBER
000106  062703  000060                                 ADD     #60,R3
000112  110322                                         MOVB    R3,(R2)+
000114  112722  000072                                 MOVB    #':,(R2)+
000120  016500  000006                                 MOV     N.FNAM(R5),R0
000124  004767  000000G                                 JSR     PC,R50OUT
000130  016500  000010                                 MOV     N.FNAM+2(R5),R0
000134  004767  000000G                                 JSR     PC,R50OUT
000140  016500  000012                                 MOV     N.FNAM+4(R5),R0
000144  004767  000000G                                 JSR     PC,R50OUT
000150  112722  000056                                 MOVB    #'.,(R2)+
000154  016500  000014                                 MOV     N.FTYP(R5),R0
000160  004767  000000G                                 JSR     PC,R50OUT
000164  112722  000073                                 MOVB    #';,(R2)+
000170  016500  000016                                 MOV     N.FVER(R5),R0
000174  004767  000000G                                 JSR     PC,R8OUT         ;CONVERT TO OCTAL
000200  062767  000025  000016G                        ADD     #21.,QDPBM+Q.IOPL+2
000206                                       40$:       DIR$    #QDPBM
                                                        .ENDC
000214  000167  000000G                     GOGET2:    JMP     CROUTX           ;PRINT PROMPT AND GET NEXT U'B ENTRY
        000001                                          .END
```

SYMBOL TABLE

```
CROUTX= ****** GX        F.EFBK= 000010        F.RATT= 000001        N.DVNM= 000032        Q.IOPR= 000007
FDBIN = ****** G         F.EFN = 000050        F.RCNM= 000034        N.FID = 000000        Q.IOSB= 000010
FDBOUO= ****** G         F.EOBB= 000032        F.RCTL= 000017        N.FNAM= 000006        RSX   = 000001
F.ACTL= 000076           F.ERR = 000052        F.RSIZ= 000002        N.FTYP= 000014        RSX11M= 000001
F.ALOC= 000040           F.FACC= 000043        F.RTYP= 000000        N.FVER= 000016        R50OUT= ****** GX
F.BBFS= 000062           F.FFBY= 000014        F.SEQN= 000100        N.NEXT= 000022        R80UT = ****** GX
F.BDB = 000070           F.FNAM= 000110        F.SPDV= 000072        N.STAT= 000020        S.FATT= 000016
F.BGBC= 000057           F.FNB = 000102        F.SPUN= 000074        N.UNIT= 000034        S.FDB = 000140
F.BKDN= 000026           F.FTYP= 000116        F.STBK= 000036        QDPBM = ****** G      S.FNAM= 000006
F.BKDS= 000020           F.FVER= 000120        F.UNIT= 000136        QEND  = ****** GX     S.FNB = 000036
F.BKEF= 000050           F.HIBK= 000004        F.URBD= 000020        QIOLBL  000000R   002 S.FNBW= 000017
F.BKP1= 000051           F.LUN = 000042        F.VBN = 000064        QLNGTH= ****** GX     S.FNTY= 000004
F.BKST= 000024           F.MBCT= 000054        F.VBSZ= 000060        QSTAT = ****** GX     S.FTYP= 000002
F.BKVB= 000064           F.MBC1= 000055        GOGET2  000214R   002 QUEHAN  000000RG   002 S.NFEN= 000020
F.CHR = 000075           F.MBFG= 000056        KBIBUF= ****** GX     Q.IOAE= 000012        TAGNUM= 000000
F.CNTG= 000034           F.NRBD= 000024        MODE  = ****** GX     Q.IOEF= 000006        $$$ARG= 000000
F.DFNB= 000046           F.NREC= 000030        NRCVSF= 000000        Q.IOFN= 000002        $$$OST= 000014
F.DSPT= 000044           F.OVBS= 000030        NTRNSF= 000002        Q.IOLU= 000004        ...TPC= 000140
F.DVNM= 000134           F.RACC= 000016        N.DID = 000024        Q.IOPL= 000014

.  ABS.   000000      000
          000000      001
MAIN      000220      002
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:   1842 WORDS  ( 8 PAGES)
DYNAMIC MEMORY:   2810 WORDS  ( 10 PAGES)
ELAPSED TIME:   00:00:19
 256,020 QUEHAN;27,  256,030 QUEHAN/CRF/-SP=UAPRXM,QUEHAN
```

QUEHAN CREATED BY MACRO ON 6-DEC-77 AT 08:27 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES				
CROUTX	= ******	GX	2-105				
FDBIN	= ******	G	2-35	2-77			
FDBOUD	= ******	G	2-35	2-82			
F.FNB	= 000102		2-77	2-82			
GOGET2	000214	R	#2-105				
KBIBUF	= ******	GX	2-68				
MODE	= ******	GX	2-78	2-80			
NRCVSF	= 000000		#1-5				
NTRNSF	= 000002		#1-4				
N.DVNM	= 000032		2-83	2-85			
N.FNAM	= 000006		2-90	2-92	2-94		
N.FTYP	= 000014		2-97				
N.FVER	= 000016		2-100				
N.UNIT	= 000034		2-86				
QDPBM	= ******	G	2-35	*2-69	*2-76	*2-102	?-103
QEND	= ******	GX	2-71				
QIOLBL	000000	R	#2-40	2-41			
QLNGTH	= ******	GX	2-76				
QSTAT	= ******	GX	2-70				
QUEHAN	000000	RG	#2-53				
Q.IOFN	= 000002		2-40				
Q.IOPL	= 000014		*2-69	*2-76	*2-102		
RSX	= 000001		#1-2	2-67			
RSX11M	= 000001		#1-3				
RT11	= ******		2-53				
R50OUT	= ******	GX	2-74	2-91	2-93	2-95	2-98
R8OUT	= ******	GX	2-101				
TAGNUM	= 000000		#1-8				
$$$ARG	= 000000		#2-40	2-40	2-40		
$$$GLB	= ******		2-40				

MACRO NAME	REFERENCES	
DIR$	#2-38	#2-103
ERROR	#1-9	
ERRORK	#1-16	
ERRORM	#1-32	
ERRORP	#1-47	
FATAL	#1-53	
FDOFF$	#2-38	2-39
OFF$	#2-40	
QDPB$	#2-40	#2-40
Q10$	#2-38	#2-40

```
RADCX   MACRO M1104  06-DEC-77 08:27  PAGE 1


                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM

                                ;
                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                                .TITLE  RADCX
                                .IDENT  /00001/
                        ;
                        ; COPYRIGHT (C)  1975, 1976, 1977
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                        ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                        ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                        ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                        ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                        ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                        ;
                        ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ; DEC.
                        ;
                        ; VERSION 0
                        ;
                        ; ROBERT H.  ROSENBAUM 2-AUGUST-74
                        ;
                        ; INCREMENT RAD50
                        ;
000000                          .PSECT  MAIN
                        ;
                        ;;;;;;;;;;;;;;;;
                        ;       ROUTINE TO INCREMENT THREE DIGITS STORED IN RAD50 BASE
                        ;       AND THEN COVERT TO RAD50
                        ;;;;;;;;;;;;;;;;
                                .IF DF  RT11
                        RADINC: MOV     #3,R2
                                MOV     #CHEXT+3,R0
                        ANTD:   INCB    -(R0)
                                .IF DF  RT11
                                CMPB    (R0),#47
                                .ENDC
                                .IF DF  RSX
                                CMPB    (R0),#70
                                .ENDC
                                BLE     RADCN
                                .IF DF  RT11
                                MOVB    #36,(R0)
                                .ENDC
                                .IF DF  RSX
                                MOV     #60,(R0)
                                .ENDC
                                INCB    -(R0)
                                DEC     R2
                                BLE     ANTD
                        RADCN:  MOV     #CHEXT,R0
                                .ENDC
```

```
     58 000000  012746  022200          RADCX::  MOV     #22200,-(SP)
     59 000004  005003                           CLR     R3
     60 000006  112005                  NEXTCHR: MOVB    (R0)+,R5        ;GET ASCII BYTE
     61                                           .IF DF  RSX
     62 000010  042705  177700                    BIC     #177700,R5
     63 000014  020527  000032                    CMP     R5,#32
     64 000020  003404                            BLE     LETT
     65 000022  020527  000060                    CMP     R5,#60
     66 000026  002401                            BLT     LETT
     67 000030  005005                  CHAR:    CLR     R5
     68 000032                          LETT:
     69                                           .ENDC
     70 000032  006303                            ASL     R3
     71 000034  006303                            ASL     R3
     72 000036  006303                            ASL     R3
     73 000040  060305                            ADD     R3,R5
     74 000042  006303                            ASL     R3
     75 000044  006303                            ASL     R3
     76 000046  060503                            ADD     R5,R3
     77 000050  006116                            ROL     (SP)            ;FINISHED?
     78 000052  103355                            BCC     NEXTCHR         ;NO
     79 000054  010300                            MOV     R3,R0
     80 000056  005726                            TST     (SP)+           ;CLEAR STACK
     81 000060  000207                            RTS     PC
     82         000000!                           .END
```

```
RADCX   MACRO M1104  06-DEC-77 08:27  PAGE 2-2
SYMBOL TABLE

CHAR    000030R     002 NEXTCH  000006R     002 NTRNSF= 000002          RSX    = 000001          TAGNUM= 000000
LETT    000032R     002 NRCVSF= 000000          RADCX   000000RG    002 RSX11M= 000001

. ABS.  000000      000
        000000      001
MAIN    000062      002
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:   380 WORDS  ( 2 PAGES)
DYNAMIC MEMORY:   2810 WORDS  ( 10 PAGES)
ELAPSED TIME:   00:00:07
 256,020 RADCX:27, 256,030 RADCX/CRF/-SP=UAPRXM,RADCX
```

RADCX CREATED BY MACRO ON 6-DEC-77 AT 08:27 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL		VALUE		REFERENCES		
CHAR		000030	R	#2-67		
LETT		000032	R	2-64	2-66	#2-68
NEXTCH		000006	R	#2-60	2-78	
NRCVSF	=	000000		#1-5		
NTRNSF	=	000002		#1-4		
RADCX		000000	RG	#2-58		
RSX	=	000001		#1-2	2-61	
RSX11M	=	000001		#1-3		
RT11	=	******		2-36		
TAGNUM	=	000000		#1-8		

MACRO NAME	REFERENCES
ERROR	#1-9
ERRORK	#1-16
ERRORM	#1-32
ERRORP	#1-47
FATAL	#1-53

```
                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM


                                ;
                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                                .TITLE  RADOUT
                                .IDENT  /00004/
                        ;
                        ;COPYRIGHT (C) 1975,1976, 1977
                        ;DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                        ;
                        ;THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY
                        ;ON A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ;THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ;SOFTWARE, OR ANY COPIES THEREOF, MAY NOT BE PROVIDED OR
                        ;OTHERWISE MADE AVBAILABLE TO ANY OTHER PERSON EXCEPT
                        ;FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO THESE
                        ;LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE SOFTWARE
                        ;SHALL AT ALL TIMES REMAIN IN DEC.
                        ;
                        ;THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ;WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ;MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ;DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ;OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ;DEC.
                        ;
                        ;
                        ; V00-02 3/26/75
                        ; V00-03 4/28/75
                        ;V00-04 9/77 CEP
                        ;MODIFICATION TO PROTECT AGAINST MORE THAN 3 OUTPUT CHARACTERS
                                .IF DF RSX
000000                          .PSECT  MAIN
                                .ENDC
                                .IF DF RT11
                                .CSECT  RADOUT
                                .ENDC
                        ;
                        ;
                                .GLOBL  R8OUT,R50OUT,R10OUT
                        ;
                        ;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                        ;       R8OUT WILL OUTPUT A RADIX 8  NUMBER TO THE CONSOLE
                        ;;;;;;;;;;;;;;;;;;;;;;;;;;;;
000000  012767  000010  000052  R8OUT:  MOV     #10,RADIX               ;SET FOR RAD 8
000006  000407                         BR      RADXX
                        ;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                        ;       R10OUT WILL OUTPUT A DECIMAL NUMBER TO THE CONSOLE
                        ;;;;;;;;;;;;;;;;;;;;;;;;;;;;
000010  012767  000012  000042  R10OUT: MOV     #10.,RADIX
000016  000403                         BR      RADXX
                        ;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                        ;       R50OUT WILL OUTPUT TO  THE CONSOLE
                        ;       THE WORD STORED AS RAD50 IN REGISTER 0
                        ;;;;;;;;;;;;;;;;;;;;;;;;;;;;
000020  012767  000050  000032  R50OUT: MOV     #50,RADIX       ;SET FOR RAD 50
000026  012703  000003          RADXX:  MOV     #3,R3           ;THREE CHARACTERS
000032  012701  000177'                 MOV     #DIGITS+3,R1    ;STORED HERE
```

```
                                CNVLUP: CLR     R4              ;REMAINDER WORD
                                        MOV     #20,-(SP)       ;BIT COUNT
                                DIVLUP: ASL     R0
                                        ROL     R4
                                        CMP     R4,RADIX        ;BIG ENOUGH TO SUBTRACT
                                        BLO     NOFIT           ;NO
                                        SUB     (PC)+,R4        ;SUBTRACT DIVSOR
                                RADIX:  +       50
                                        INC     R0              ;CORRESPONDING BIT IN QUOTEINT
                                NOFIT:  DEC     (SP)
                                        BNE     DIVLUP          ;MORE BITS
                                        TST     (SP)+           ;CLEAR STACK
                                        DEC     R3              ;ARE WE DONE?
                                        EMI     R50LUP          ;YES - HAVE DONE 3
                                        MOVB    R4,-(R1)        ;STORE CHARACTER
                                        TST     R0              ;MORE
                                        BNE     CNVLUP
                                BLFILL: DEC     R3              ;PAD WITH SPACE
                                        BMI     R50LUP          ;NO
                                        MOVB    #40,(R2)+
                                        BR      BLFILL
                                R50LUP: MOVB    (R1)+,R0        ;GET CHARACTER
                                        BMI     DONE            ;FINISHED
                                        CMP     RADIX,#50       ;A NUMBER?
                                        BNE     NUMB            ;YES
                                        TSTB    R0              ;A SPACE?
                                        BEQ     PSPACE          ;YES
                                        CMPB    R0,#32          ;A LETTER
                                        BLE     LETTER
                                        SUB     #36,R0          ;A DIGIT
                                        BCC     NUMB            ;A NUMBER
                                PSPACE: MOV     #-40,R0         ;NO
                                LETTER: ADD     #20,R0
                                NUMB:   ADD     #60,R0
                                        MOVB    R0,(R2)+
                                        BR      R50LUP
                                DONE:   RTS     PC
                                DIGITS: .BYTE   0,0,0,377
                                        .EVEN

                                        .END
```

```
RADOUT  MACRO M1104  06-DEC-77 08:27  PAGE 2-2
SYMBOL TABLE

BLFILL  000104R     002 DONE    000172R     002 NTRNSF= 000002          RADXX   000026R     002 R50LUP  000116R     002
CNVLUP  000036R     002 LETTER  000156R     002 NUMB    000162R     002 RSX   = 000001          R50OUT  000020RG    002
DIGITS  000174R     002 NOFIT   000064R     002 PSPACE  000152R     002 RSX11M= 000001          R80UT   000000RG    002
DIVLUP  000044R     002 NRCVSF= 000000          RADIX   000060R     002 R100UT  000010RG    002 TAGNUM= 000000

. ABS.  000000      000
        000000      001
MAIN    000200      002
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  445 WORDS  ( 2 PAGES)
DYNAMIC MEMORY:  2810 WORDS  ( 10 PAGES)
ELAPSED TIME:  00:00:07
 256,020 RADOUT;27, 256,030 RADOUT/CRF/-SP=UAPRXM,RADOUT
```

```
RADOUT     CREATED BY  MACRO  ON 6-DEC-77 AT 08:27      PAGE 1
SYMBOL CROSS REFERENCE                                  CREF
SYMBOL   VALUE              REFERENCES
BLFILL     000104 R         #2-75        2-78
CNVLUP     000036 R         #2-58        2-74
DIGITS     000174 R          2-57       #2-95
DIVLUP     000044 R         #2-60        2-68
DONE       000172 R          2-80       #2-94
LETTER     000156 R          2-86       #2-90
NOFIT      000064 R          2-63       #2-67
NRCVSF   = 000000           #1-5
NTRNSF   = 000002           #1-4
NUMB       000162 R          2-82        2-88       #2-91
PSPACE     000152 R          2-84       #2-89
RADIX      000060 R         *2-42       *2-48      *2-55       2-62       #2-65       2-81
RADXX      000026 R          2-43        2-49       #2-56
RSX      = 000001           #1-2         2-29
RSX11M   = 000001           #1-3
RT11     = ******            2-32
R10OUT     000010 RG         2-37       #2-48
R50LUP     000116 R          2-71        2-76       #2-79       2-93
R50OUT     000020 RG         2-37       #2-55
R8OUT      000000 RG         2-37       #2-42
TAGNUM   = 000000           #1-8
```

RADOUT CREATED BY MACRO ON 6-DEC-77 AT 08: 27 PAGE 2
MACRO CROSS REFERENCE CREF

MACRO NAME	REFERENCES
ERROR	#1-9
ERRORK	#1-16
ERRORM	#1-32
ERRORP	#1-47
FATAL	#1-53

READ MACRO M1104 06-DEC-77 08:27 PAGE 1

```
                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM


                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                                .TITLE  READ
                                .IDENT /00004/
                        ;
                        ; COPYRIGHT (C)  1975 1976, 1977
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                        ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                        ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                        ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                        ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                        ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                        ;
                        ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ; DEC.
                        ;
                        ;
                        ; VERSION 0
                        ;
                        ; ROBERT H. ROSENBAUM
                        ;
                        ; LAST MODIFICATION ON 26-MARCH-75
                        ;
                        ; READ A BUFFER IN MOVE MODE
                        ;
000000                          PSECT  MAIN
                        ;
                        ; MACRO CALLS
                                .MCALL  GET$S,FDOFF$
000000                          FDOFF$  DEF$L           ;DEFINE FCS OFFSETS LOCALLY
                        ;
                        ; ASCII DEFINITIONS
                                ETX=3
                                EM=31
                                ESC=33
                        ;
                        ;+
                        ; INPUTS:
                        ;       R0 = ADDRESS OF BUFFER
                        ;
                        ; OUTPUTS:
                        ;       BUFFER POINTED TO BY R0 IS FILLED
                        ;       THE ROUTINE FIRST MOVES THE DATA FROM FCS'S BUFFER TO ONE
                        ;       OF ITS OWN 7 BUFFERS WHICH IT WILL EVENTUALLY GIVE TO
                        ;       THE TAP TO TRANSMIT.  IT WILL ONLY DO A "GET" TO FCS UPON
                        ;       TRANSFERRING THE LAST CHARACTER FROM INPUT RECORD.
                        ;       ERRORS ARE NOT CHECKED AFTER THE GET BUT UPON THE NEXT
                        ;       CALL TO RED$ SINCE IT IS ONLY AT THIS TIME THAT THE DATA IN
                        ;       ERROR WILL BE PROCESSED
                        ;-
```

```
READ    MACRO M1104  06-DEC-77 08:27  PAGE 2-1


                                        .IF DF  RSX
000000  010001                REDS::  MOV     R0,R1           ;BEGINNING OF BUFFER
000002  010005                        MOV     R0,R5           ;SAVE IT
000004  012702  000000G               MOV     #FDBIN,R2       ;FDB ADDDRESS
000010  105762  000052                TSTB    F.ERR(R2)       ;ERROR ON GET
000014  100450                        BMI     REDERR
000016  016203  000026        REDSUM: MOV     F.NRBD+2(R2),R3 ;BLOCK BUFFER ADDRESS
000022  016202  000024                MOV     F.NRBD(R2),R2   ;BYTE COUNT
000026  060002                        ADD     R0,R2           ;END OF RECORD
000030  066700  000000G               ADD     RECSIZ,R0       ;EDN OF RECORD FOR TRANSMIT
000034  000421                        BR      REDTRN
000036  132767  000002  000000G REDSPL: BITB   #2,FLAGS       ;TRANSPARENT MODE
000044  001015                        BNE     REDTRN          ;YES
000046  120427  000003                CMPB    R4,#ETX         ;IS IT AN ETX?
000052  001405                        BEQ     REDETX          ;YES
000054  120427  000031                CMPB    R4,#EM          ;IS IT AN EM
000060  001007                        BNE     REDTRN          ;NO
000062  005301                        DEC     R1
000064  000461                        BR      REXIT
000066  005301                REDETX: DEC     R1
000070  152767  000200  000000G       BISB    #200,ENDPEN
000076  000407                        BR      REDAG
000100  020102                REDTRN: CMP     R1,R2           ;FINISHED? (END OF RECORD)
000102  001405                        BEQ     REDAG           ;YES
000104  020100                        CMP     R1,R0           ;EQUAL TO TRANSMIT BUFFER SIZE
000106  001450                        BEQ     REXIT           ;YES, DON'T READ ANY MORE
000110  112304                        MOVB    (R3)+,R4        ;GET CHARACTER
000112  110421                        MOVB    R4,(R1)+        ;MOVE IT TO TRANSMIT BUFFER
000114  000750                        BR      REDSPL
000116  105767  000000G         REDAG: TSTB    MCB             ;GENERAL BLOCKING MODE?
000122  001042                        BNE     REXIT           ;YES, DON'T PAD THE RECORD
000124  020100                REDCL:  CMP     R1,R0           ;TRANSMIT BUFFER FULL?
000126  001447                        BEQ     RGET            ;YES, NO NEED TO PAD IT
000130  112721  000040                MOVB    #40,(R1)+       ;NO, PAD WITH ASCII SPACE
000134  000773                        BR      REDCL           ; UNTIL THE BUFFER IS FILLED
                                ;
                                ; CONTROL COMES TO REDERR IF THERE WAS AN ERROR FROM THE PREVIOUS GET
                                ;
000136  122762  000000G 000052 REDERR: CMPB   #IE.RBG,F.ERR(R2)       ;RECORD TOO BIG
000144  001007                        BNE     REDEOF
000146                                ERROR   <RECORD TOO LONG>
000162  000715                        BR      REDSUM
000164  122762  000000G 000052 REDEOF: CMPB   #IE.EOF,F.ERR(R2)       ;END OF FILE?
000172  001434                        BEQ     RFNL            ;YES
000174                                ERROR   <IN ERR>        ;NO ANOTHER TYPE OF ERROR
000210  112767  000200  000000G       MOVB    #200,ELATCH     ;SET ETX LATCH
000216  112767  000200  000000G       MOVB    #200,IOERR      ;INDICATE I/O ERROR
000224  000261                        SEC                     ;ERROR RETURN INDICATOR
000226  000207                        RTS     PC              ;AND RETURN
000230  160102                REXIT:  SUB     R1,R2
000232  001405                        BEQ     RGET
000234  010267  000024G               MOV     R2,F.NRBD+FDBIN
000240  010367  000026G               MOV     R3,F.NRBD+2+FDBIN
000244  000407                        BR      RFNL
000246  105767  000000G         RGET: TSTB    RVIPEN          ;RVI PENDING?
000252  100404                        BMI     RFNL            ;YES, DON'T DO ANOTHER GET
000254                                GET$S   #FDBIN,,
```

```
    115 000264  160501                RFNL:   SUB     R5,R1
    116 000266  000241                        CLC                     ;INDICATE GOOD RETURN
    117 000270  000207                        RTS     PC
    118                                       .ENDC
    119         000001                        .END
```

```
READ    MACRO M1104  06-DEC-77 08:27  PAGE 2-3
SYMBOL TABLE

ELATCH= ****** GX       F.DSPT= 000044          F.RATT= 000001          N.FID = 000000          RGET    000246R     002
EM    = 000031          F.DVNM= 000134          F.RCNM= 000034          N.FNAM= 000006          RSX   = 000001
ENDPEN= ****** GX       F.EFBK= 000010          F.RCTL= 000017          N.FTYP= 000014          RSX11M= 000001
ESC   = 000033          F.EFN = 000050          F.RSIZ= 000002          N.FVER= 000016          RVIPEN= ****** GX
ETX   = 000003          F.EOBB= 000032          F.RTYP= 000000          N.NEXT= 000022          S.FATT= 000016
FDBIN = ****** GX       F.ERR = 000052          F.SEQN= 000100          N.STAT= 000020          S.FDB = 000140
FLAGS = ****** GX       F.FACC= 000043          F.SPDV= 000072          N.UNIT= 000034          S.FNAM= 000006
F.ACTL= 000076          F.FFBY= 000014          F.SPUN= 000074          PAR$$$= 000027          S.FNB = 000036
F.ALOC= 000040          F.FNAM= 000110          F.STBK= 000036          PROMPT= ****** GX       S.FNBW= 000017
F.BBFS= 000062          F.FNB = 000102          F.UNIT= 000136          RECSIZ= ****** GX       S.FNTY= 000004
F.BDB = 000070          F.FTYP= 000116          F.URBD= 000020          REDAG   000116R     002 S.FTYP= 000002
F.BGBC= 000057          F.FVER= 000120          F.VBN = 000064          REDCL   000124R     002 S.NFEN= 000020
F.BKDN= 000026          F.HIBK= 000004          F.VBSZ= 000060          REDEOF  000164R     002 TAGE1   000017R     003
F.BKDS= 000020          F.LUN = 000042          IE.EOF= ****** GX       REDERR  000136R     002 TAGE2   000025R     003
F.BKEF= 000050          F.MBCT= 000054          IE.RBG= ****** GX       REDETX  000066R     002 TAGNUM= 000002
F.BKP1= 000051          F.MBC1= 000055          IOERR = ****** GX       REDSPL  000036R     002 TAG1    000000R     003
F.BKST= 000024          F.MBFG= 000056          MCB   = ****** GX       REDSUM  000016R     002 TAG2    000017R     003
F.BKVB= 000064          F.NRBD= 000024          NRCVSF= 000000          REDTRN  000100R     002 TYPMSG= ****** GX
F.CHR = 000075          F.NREC= 000030          NTRNSF= 000002          RED$    000000RG    002 .GETSQ= ****** G
F.CNTG= 000034          F.OVBS= 000030          N.DID = 000024          REXIT   000230R     002 ...TPC= 000140
F.DFNB= 000046          F.PACC= 000016          N.DVNM= 000032          RFNL    000264R     002

. ABS.   000000     000
         000000     001
MAIN     000272     002
MESAG2   000025     003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  1859 WORDS  ( 8 PAGES)
DYNAMIC MEMORY:  2810 WORDS  ( 10 PAGES)
ELAPSED TIME:  00:00:19
 256,020 READ;27, 256,030 READ/CRF/-SP=UAPRXM,READ
```

READ CREATED BY MACRO ON 6-DEC-77 AT 08:27 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL		VALUE		REFERENCES					
ELATCH	=	******	GX	*2-103					
EM	=	000031		#2-41	2-73				
ENDPEN	=	******	GX	*2-78					
ESC	=	000033		#2-42					
ETX	=	000003		#2-40	2-71				
FDBIN	=	******	GX	2-61	*2-109	*2-110	2-114	2-114	
FLAGS	=	******	GX	2-69					
F.ERR	=	000052		2-62	2-96	2-100			
F.NRBD	=	000024		2-64	2-65	*2-109	*2-110		
IE.EOF	=	******	GX	2-100					
IE.RBG	=	******	GX	2-96					
IOERR	=	******	GX	*2-104					
MCB	=	******	GX	2-87					
NRCVSF	=	000000		#1-5					
NTRNSF	=	000002		#1-4					
PARSSS	=	000027		#2-114	2-114				
PROMPT	=	******	GX	2-98	2-102				
RECSIZ	=	******	GX	2-67					
REDAG		000116	R	2-79	2-81	#2-87			
REDCL		000124	R	#2-89	2-92				
REDEOF		000164	R	2-97	#2-100				
REDERR		000136	R	2-63	#2-96				
REDETX		000066	R	2-72	#2-77				
REDSPL		000036	R	#2-69	2-86				
REDSUM		000016	R	#2-64	2-99				
REDTRN		000100	R	2-68	2-70	2-74	#2-80		
REDS		000000	RG	#2-59					
REXIT		000230	R	2-76	2-83	2-88	#2-107		
RFNL		000264	R	2-101	2-111	2-113	#2-115		
RGET		000246	R	2-90	2-108	#2-112			
RSX	=	000001		#1-2	2-58				
RSX11M	=	000001		#1-3					
RVIPEN	=	******	GX	2-112					
TAGE1		000017	R	2-98	#2-98				
TAGE2		000025	R	2-102	#2-102				
TAGNUM	=	000002		#1-8	2-98	#2-98	2-98	2-102	#2-102	2-102
TAG1		000000	R	2-98	2-98	#2-98			
TAG2		000017	R	2-102	2-102	#2-102			
TYPMSG	=	******	GX	2-98	2-102				
.GETSQ	=	******	G	2-114	2-114				

MACRO NAME	REFERENCES		
CMOV$W	#2-114	#2-114	#2-114
ERROR	#1-9	2-98	2-102
ERRORK	#1-16	2-98	2-102
ERRORM	#1-32		
ERRORP	#1-47		
ERR$	#2-114	#2-114	
FATAL	#1-53		
FDOFF$	#2-36	#2-37	
GET$S	#2-36	2-114	
LDFDB$	#2-114	2-114	

COMTEX-MACROS MACRO M1104 06-DEC-77 08:28
TABLE OF CONTENTS

3- 21 COMTEX MACRO DEFINITIONS
5- 70 COMMUNICATIONS LINE CODE
6- 139 OPEN FILES FOR RECEIVE
7- 266 VERTICAL FORMS CONTROL PROCESSING
8- 311 HORIZONTAL FORMS CONTROL
9- 332 CARRIAGE CONTROL
11- 367 ASSIGN BUFFER TO BYSYNC TAP
12- 382 TRANSLATION ROUTINE FOR PRINT DATA (/BI/SL)
13- 399 MOVE DATA TO FNB OR COMMON SWITCH AREA
14- 414 CONSTANT DATA AREA

RECEIVE MACRO M1104 06-DEC-77 08:28 PAGE 1

```
                        ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                  RSX=1
000001                  RSX11M=1
000002                  NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                  NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                        ;
                        ;MACRO DEFINITIONS
000000                          TAGNUM=0
                                .MACRO  ERROR   TEXT
                                ERRORK  <TEXT>
                                JSR     PC,PROMPT
                                .ENDM
                        ;
                        ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                        ;
                                .MACRO  ERRORK  TEXT
                                TAGNUM=TAGNUM+1
                                .IRP    N,<TAGNUM>
                                JSR     R5,TYPMSG
                                .WORD   TAG'N
                                .WORD   TAGE'N-TAG'N
                                .PSECT  MESAG2,GBL
                        TAG'N:  .ASCII  "TEXT"
                        TAGE'N:
                                .PSECT  MAIN
                                .ENDM
                                .ENDM

                        ;
                        ; MACRO TO QUEUE A MESSAGE ONLY
                        ;
                                .MACRO  ERRORM  TEXT
                                TAGNUM=TAGNUM+1
                                .IRP    N,<TAGNUM>
                                JSR     R5,TYPMS1
                                .WORD   TAG'N
                                .WORD   TAGE'N-TAG'N
                                .PSECT  MESAG2,GBL
                        TAG'N:  .ASCII  "TEXT"
                        TAGE'N:
                                .PSECT  MAIN
                                .ENDM
                                .ENDM

                        ; MACRO TO QUEUE A MESSAGE AND PROMPT
                        ;
                                .MACRO  ERRORP  TEXT
                                ERRORM  <TEXT>
                                JSR     PC,PROMPT
                                .ENDM

                        ;
                                .MACRO  FATAL   TEXT
                                ERRORK  <TEXT>
                                JSR     R5,DIE
                                .ENDM
```

```
                        .TITLE  COMTEX-MACROS
                        .IDENT  /00002/
                ;
                ;COPYRIGHT 1974, 1975, 1976, 1977
                ;DIGITAL EQUIPMENT CORP.
                ;146 MAIN STREET          MAYNARD, MASS. 01754
                ;
                ;DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR
                ;RELIABILITY OF ITS SOFTWARE ON EQUIPMENT WHICH
                ;IS NOT SUPPLIED BY DIGITAL
                ;
                ; VERSION 0
                ;
                ; 23-SEPTEMBER-74
                ;
                ;VERSION 1
                ; SEP 1977 - CEP
                ; MODIFICATIONS FOR CUT CODE VERSION OF 2780
                ; LIBRARY OF COMTEX MACRO DEFINITIONS
```

```
COMTEX-MACROS   MACRO M1104  06-DEC-77 08.28  PAGE 3
COMTEX MACRO DEFINITIONS

                                        .SBTTL  COMTEX MACRO DEFINITIONS
                                ;
                                        .MACRO  LINIT,PARLST
                                        MOV     #PARLST,R1
                                        JSR     PC,EMTINT
                                        .ENDM
                                ;
                                        .MACRO  ASRBUF,PARLST
                                        MOV     #PARLST,R1
                                        JSR     PC,EMTPUT
                                        .ENDM
                                ;
                                        .MACRO  PUTTC,PARLST
                                        MOV     #PARLST,R1
                                        JSR     PC,EMTPUT
                                        .ENDM
                                ;
                                        .MACRO  PUTMC,PARLST
                                        MOV     #PARLST,R1
                                        JSR     PC,EMTPUT
                                        .ENDM
                                ;
                                        .MACRO  GETS
                                        JSR     PC.EMTGET
                                        .ENDM

                                        .MACRO  PUTTM,PARLST
                                        MOV     #PARLST,R1
                                        JSR     PC,EMTTIM
                                        .ENDM

                                        .MACRO  PUTD,PARLST
                                        MOV     #PARLST,R1
                                        JSR     PC.EMTPUT
                                        .ENDM
```

```
                        .TITLE  RECEIVE
                        .IDENT  /00023/
                ;
                ; COPYRIGHT (C)  1975, 1976, 1977
                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                ;
                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                ;
                ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                ;
                ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                ; DEC.
                ;
                ; VERSION 0
                ;
                ; ROBERT H. ROSENBAUM 20-SEPTEMBER-74
                ;
                ; V00-2 10/1/74
                ; V00-3 11/13/74
                ; V00-4 11/20/74
                ; V00-5 1/16/75
                ; V00-6 1/17/75
                ; V00-7 2/13/75
                ; V00-10 2/19/75
                ; V00-11 2/24/75
                ; V00-12 3/26/75
                ; V00-13 3/31/75
                ; V00-14 4/28/75
                ; V00-15 5/20/75
                ; V00-16 5/28/75
                ; V00-17 6/11/75
                ; V00-20 6/23/75
                ; V00-21 6/27/75
                ; V00-22 7/3/75
                ;V00-23 9/77 - CEP
                ;PUT IN FIX FOR 'IDLE-LOOP'  'ROBLEM
                ;PUT IN FIX FOR RECEIVE HANC PROBLEM
                ;PUT IN FIX FOR PUNCH FILE FROBLEM
                ;
                ;+
                ;RECEIVE DATA COMPLETE ROUTINE
                ;
                ; CALL:
                ;       INPCPL IS CALLED FROM THE "GETS" LOOP THROUGH THE JUMP TABLE
                ;       UPON FINDING A UIB ENTRY FROM THE BISYNC TAP
                ;
                ; FUNCTION:
```

```
                                ;       THIS ROUTINE WRITES THE DATA TO THE OUTPUT DEVICE TAKING CARE
                                ;       OF ALL VERTICAL AND HORIZONTAL FORMATING
                                ;-
000000                                  .PSECT  MAIN
                                ;
                                ; MACRO CALLS
                                        .MCALL  CLOSE$
                                ;
                                ; RECEIVE STATUS BIT DEFINITIONS
001000                                  TRNREC=1000             ;TRANSPARENT RECORD
000020                                  UOVR=20                 ;USER BUFFER OVERFLOW
```

```
RECEIVE MACRO M1104  06-DEC-77 08:28  PAGE 5
COMMUNICATIONS LINE CODE

                                        .SBTTL  COMMUNICATIONS LINE CODE

                                ;COMMUNICATIONS INPUT COMPLETE

                                        .ENABL  LSB
000000  105767  000000G        INPCPL::TSTB    ONLINE          ;ONLINE
000004  100112                        BPL     GOASR           ;REASSIGN BUFFER
000006  012705  000006G                MOV     #ENTRY+6,R5
000012  012704  000002G                MOV     #ENTRY+2,R4     ;ADDRESS OF DATA BUFFER
000016  032715  006000                BIT     #6000,(R5)              ;IS IT AN EOT OR A DLE-EOT?
000022  001511                        BEQ     INCP2
000024  105267  000000G                INCB    CMBFAV          ;RELEASE BUFFER


000030  010546                        MOV     R5,-(SP)        ;SAVE ERROR BITS - MSSG ROUTINE CLOBBBERS IT
000032  106367  000000G                ASLB    ETXFLG          ;ETX RECEIVED?
000036  103425                        BCS     20$             ;YES
000040  105767  000000G                TSTB    FLAGS           ;TRANSMIT ACTIVE?
000044  100003                        BPL     10$             ;NO
000046  012605                        MOV     (SP)+,R5        ;MAKE SURE R5 RESTORED
000050  000167  000000G                JMP     SNDEOT          ;ABORT THE TRANSMISSION
000054  122767  000001  000000G 10$:  CMPB    #1,MODE         ;HAVE WE OPENED RECEIVE FILE?
000062  001036                        BNE     35$             ;NO, MUST BE CONTROL MODE
000064  106367  000000G                ASLB    OPNFLG          ;****IF NO DATA DON'T ISSUE
000070  103425                        BCS     33$             ;****MESSAGE
000072  152767  000200  000000G        BISB    #200,INCRCV     ;INCOMPLETE RECEIVE - NOTE SO THAT
                                                              ;ON NEXT RECEIVE WE WILL OPEN SAME FILE
                                                              ;PLUS ONE VERSION
                                                              ;CLEAR FLAG AT ENTER TIME
000100                                  ERRORK  <INCOMPLETE RECV> ;TYPE ERROR MESSAGE
000110  000404                        BR      30$             ;SEE IF DISCONNECT OR RVI
000112                          20$:    ERRORK  <RECV COMPLETE> ;INDICATE SUCCESSFUL COMPLETION
000122  004767  000000G        30$:    JSR     PC,CLOSER               ;RELEASE THE DATASETS
000126  005767  000000G                TST     UBFCNT          ;ANY BUFFER OVERRUNS THIS TIME?
000132  001402                        BEQ     32$             ;NO
000134  004767  000000G                JSR     PC,UBFHAN       ;YES, TELL THE OPERATOR HOW MANY
000140  004767  000000G        32$:    JSR     PC,PROMPT       ;REPROMPT
                                                              ;*******
000144  106367  000000G        33$:    ASLB    RVIPEN          ;CLEAR AND TEST IF RVI IS PENDING
000150  103003                        BCC     35$             ;NO
000152  152767  000200  000000G        BISB    #200,TXPEN      ;YES, INDICATE TRANSMIT PENDING
000160  012605                35$:    MOV     (SP)+,R5        ;RESTORE ERROR BITS
000162  032715  004000                BIT     #4000,(R5)      ;DLE-EOT?
000166  001002                        BNE     50$             ;YES
000170  000167  000000G        40$:    JMP     TURNAR
000174                          50$:    ERRORK  <DISCONNECT>
000204  000167  000000G        GODSC:  JMP     OFFHAN          ;GO OFFLINE
                                        .DSABL  LSB


000210  112767  000001  000000G NOKE:   MOVB    #1,MODE         ;SET RECEIVE MODE
000216  016767  001304  000000G        MOV     QRECE,QMODE     ;SET UP FOR QUERY TYPEOUT
000224  152767  000000G 000000G        BISB    #ACK,RESP       ;SEND AN ACK
000232  105267  000000G        GOASR:  INCB    CMBFAV          ;MAKE THE BUFFER AVAILABLE
000236  004767  001152                JSR     PC,ASROUT       ;ASSIGN THE BUFFER TO THE TAP
000242  000167  000000G        INPEXT: JMP     GETENT          ;GET THE NEXT UIB ENTRY


000246  105767  000000G        INCP2:  TSTB    FLAGS
```

```
    127 000252  100767                          BMI     GOASR           ;IF MASTER, IGNORE
    128 000254  105767  000000G                 TSTB    INITF           ;INITIALIZATION?
    129 000260  100115                          BPL     INPRA           ;NO, GO HANDLE TFR
    130 000262  005765  177776                  TST     -2(R5)          ;ANY CHARS IN BUFFER?
    131 000266  001117                          BNE     RV1ERR          ;YES - TYPE ERROR MESSAGE
    132 000270  032715  004000                  BIT     #4000,(R5)      ;CHECK FOR VALID ENQ
    133 000274  001343                          BNE     GODSC           ;NO - DISCONNECT
    134 000276  126727  000000G 000001         CMPB    MODE,#1                                          :
    135 000304  001752                          BEQ     GOASR           ;EXTRA SELECT
    136 000306  152767  000200  000000G         BISB    #200,OPNFLG     ;*****SET FLAG TO SHOW ENQ RECEIVED
    137 000314  000735                          BF      NOKE            ;******
```

```
RECEIVE MACRO M1104  06-DEC-77 08:28  PAGE 6
OPEN FILES FOR RECEIVE

                                                .SBTTL  OPEN FILES FOR RECEIVE
                                                
                                                
000316  012702  000000G         SLAVC1: MOV     #RCVBLK,R2
000322  012701  000000G                 MOV     #S.FNBW,R1      ;# OF WORDS TO MOVE TO FNB
000326  012700  000000C                 MOV     #FDBOUO+F.FNB,R0        ;ADDR OF WHERE TO MOVE THE DATA
000332  005732                         TST     @(R2)+          ;USE DEFAULT FILE NAME?
000334  100412                         BMI     DFLT            ;YES
000336  014202                         MOV     -(R2),R2        ;NO, GET POINTER TO RECEIVE NAME BLOCK
000340  004767  001152                 JSR     PC,XFRFNB       ;TRANSFER DATA FROM RCVBLK AREA TO FNB IN FDB
000344  012701  000000C                 MOV     #RCVSWT/2,R1    ;# OF WORDS OF RECEIVE SWITCHES
000350  012700  000000G                 MOV     #SWRBN,R0       ;ADDR OF FIRST SWITCH
000354  004767  001136                 JSR     PC,XFRFNB       ;TRANSFER SWITCHES TO COMMON AREA
                                                                ; FROM END OF RCVBLK FOR THIS FILE
000360  000427                         BR      DFCNT           ;OPEN THE FILE
000362  010262  177776         DFLT:   MOV     R2,-2(R2)       ;RESET RCVBLK POINTER TO START
000366  012712  177777                 MOV     #-1,(R2)        ;INDICATE NO RECV CMMDS PENDING
                                        .IF DF  RT11
                                        MOV     DEVLP,(R2)+     ;DEFAULT TO LP
                                        CLR     (R2)+           ;FILENAME
                                        CLR     (R2)+           ;FILENAME
                                        CLR     (R2)+           ;EXTENSION
                                        CLR     (R2)+           ;SIZE
                                        CLR     (R2)+           ;END INDICATOR
                                        CLR     RCVSW
                                        BR      SLAVC1
                                        .ENDC
                                        .IF DF  RSX
000372  012702  000000G                 MOV     #DFNLP,R2       ;SINCE NO RECEIVE CMMD, DEFAULT
000376  004767  001114                 JSR     PC,XFRFNB       ;TRANSFER DATA FROM DEFAULT NAME BLOCK
                                                                ; TO FNB IN FDB
000402  005067  000000G                 CLR     SWRBN           ;CLEAR ALL SWITCHES
000406  005067  000000G                 CLR     SWRSL           ;
000412  005067  000000G                 CLR     RCHN            ;
000416  012767  000206  000000G         MOV     #134.,RVAL      ;
000424  012767  000000G 000000G         MOV     #R.VAR,RTYP     ;
000432  012767  000000G 000000G         MOV     #FD.CR,RATT     ;
000440                          DFCNT:
                                        .ENDC
                                        .IF DF  RT11
                                NDFLT:  MOV     #RCVSW,R1       ;DEFINE SWITCHES
                                        JSR     PC,SWITCHES     ;
                                        BMI     SWITOK          ;SWITCHES ARE OK(LEGAL)
                                        JMP     RERR
                                SWITOK: JSR     PC,DFNBUF
                                        .ENDC
                                        .IF DF  RSX
000440  012767  000000G 000000G         MOV     #COLTT,COLP1    ;FOR VERTICAL FORMAT CONTROL
000446  012767  000000G 000000G         MOV     #COLTT,COLP2    ;FOR VERTICAL FORMAT CONTROL
000454  005067  000000G                 CLR     RECS1           ;CLEAR TOTAL RECORD SIZE
                                        .ENDC
000460  105067  000000G                 CLRB    INCRCV          ;CLEAR INCOMPLETE RECEIVE FLAG
                                                                ;(EOT WITH NO ETX)
000464  004767  000000G                 JSR     PC,ENTER        ;ENTER  FILE
000470  103011                         BCC     INPRA           ;ERROR FROM OPEN?*********
000472                                  ERROR   <RECV OPEN ERR> ;YES - INFORM OPERATOR
000506  152767  000200  000000G         BISB    #200,IOERR      ;*******SET TO ABORT RECEIVE
```

```
RECEIVE MACRO M1104  06-DEC-77 08:28  PAGE 6-1
OPEN FILES FOR RECEIVE

    196
    197
    198 000514                                  INPRA:
    199 000514  106367  000000G                         ASLB    OPNFLG          ;*****DO WE NEED TO OPEN FILE?
    200 000520  103676                                  BCS     SLAVC1          ;*****YES
    201 000522  005715                                  TST     (R5)            ;ABNORMAL FORMAT?
    202 000524  100003                                  BPL     TSTPFL          ;NO, GO CHECK CONDITIONS
    203
    204 000526  052715  000200              RV1ERR:    BIS     #200,(R5)       ;FORCE RECORD ERROR FLAG
    205 000532  000414                                  BR      SETBW
    206
    207
    208 000534  032715  010000              TSTPFL:    BIT     #10000,(R5)     ;NO, ETX SEEN?
    209 000540  001404                                  BEQ     INPR2           ;NO, CHECK FOR AB END
    210 000542  152767  000200  000000G                BISB    #200,ETXFLG     ;YES, SET ETX FLAG
    211 000550  000405                                  BR      SETBW           ;GO TO COMMON PRINT CODE
    212 000552  032715  040000              INPR2:     BIT     #40000,(R5)     ;AB END?
    213 000556  001402                                  BEQ     SETBW           ;NO, GO COMMON CODE
    214 000560  012715  030377                          MOV     #30377,(R5)     ;SET ERROR FLAGS AND ETX FLAG
    215 000564  032715  030000              SETBW:     BIT     #30000,(R5)     ;ETX OR ETB??
    216 000570  001403                                  BEQ     SETB5           ;NO
    217 000572  152767  000200  000000G                BISB    #200,ACKPEN
    218 000600  105767  000000G             SETB5:     TSTB    IOERR           ;A WRITE ERROR?
    219 000604  001417                                  BEQ     SETB0           ;NO
    220 000606  105267  000000G                         INCB    INPCNT          ;YES - COUNT RECORD AS RECEIVED
    221 000612  105267  000000G                         INCB    CMBFAV          ;MAKE THE BUFFER AVAILABLE
    222 000616  004767  000572                          JSR     PC,ASROUT       ;ASSIGN THE BUFFER TO THE TAP
    223 000622  106367  000000G                         ASLB    ACKPEN          ;RECORD CONTAIN ETB OR ETX?
    224 000626  103205                                  BCC     INPEXT          ;WAIT FOR REMAINING RECORDS IN BLOCK
    225 000630  105067  000000G                         CLRB    IOERR           ;CLEAR ERROR
    226 000634  105067  000000G                         CLRB    INPCNT          ;CLEAR INPUT COUNT
    227 000640  000167  000000G                         JMP     SNDEOT          ;SEND EOT
    228
    229 000644  005767  000000G             SETB0:     TST     NAKCNT          ;ERROR BLOCK BEING RECEIVED?
    230 000650  100433                                  BMI     SETB4           ;YES -
    231 000652  001417                                  BEQ     SETB2           ;NOT A RETRANSMIT RECORD EITHER
    232 000654  005367  000000G                         DEC     NAKCNT          ;THROW RETRANSMITTED RECORDS AWAY
    233 000660  106367  000000G                         ASLB    ACKPEN          ;MAKE SURE IS NOT FINAL RECORD
    234 000664  103010                                  BCC     5$              ;IS OK
    235 000666  105067  000000G                         CLRB    ETXFLG          ;IS A FINAL RECORD - CLEAR ETX FLAG
    236 000672  016767  000000G 000000G                 MOV     INPCNT,NAKCNT   ;RESET NAK COUNT FOR NEXT TIME
    237 000700  152767  000000G 000000G                 BISB    #NAK,RESP       ;SET TO SEND A NAK
    238 000706  000167  177320              5$:        JMP     GOASR           ;UNTIL NAKCNT BECOMES ZERO
    239
    240 000712  105715                     SETB2:     TSTB    (R5)            ;ERROR INDICATED FOR THIS RECORD?
    241 000714  100045                                  BPL     SETB1
    242 000716  042715  000220                          BIC     #200!UOVR,(R5)  ;JUST A BUFFER OVERRUN?
    243 000722  105715                                  TSTB    (R5)            ;..
    244 000724  001003                                  BNE     10$             ;NO, OTHER ERRORS TOO
    245 000726  005267  000000G                         INC     UBFCNT          ;YES, INCREMENT # OF SUCH ERRORS
    246 000732  000436                                  BR      SETB1           ;PROCESS TRUNCATED RECORD NORMALLY
    247 000734                              10$:
    248 000734  005237  000000G                         INC     @#STKSET        ;INCREMENT BCC ERROR COUNT
    249 000740  105067  000000G             SETB4:     CLRB    ETXFLG          ;ERROR DETECTED IN BLOCK. DON'T
    250                                                                         ;ALLOW ETX ACTIONS
    251 000744  105267  000000G                         INCB    INPCNT          ;COUNT RECORD AS RECEIVED
    252 000750  105267  000000G                         INCB    NAKCNT          ;INCREMENT ERROR RECORD COUNT
```

OPEN FILES FOR RECEIVE

```
    253 000754  052767  100000  000000G          BIS     #100000,NAKCNT  ;SET RCVING BAD BLOCK FLAG
    254 000762  106367  000000G                  ASLB    ACKPEN          ;ETB OR ETX SEEN?
    255 000766  103402                           BCS     SETB3           ;YES
    256 000770  000167  177236                   JMP     GOASR           ;THROW RECORD AWAY
    257
    258 000774  042767  100000  000000G SETB3:   BIC     #100000,NAKCNT  ;END OF ERROR BLOCK
    259 001002  166767  000000G 000000G          SUB     NAKCNT,INPCNT   ;NUMBER OF GOOD RECORDS IN BLOCK
    260                                                                  ;THAT WERE PRINTED
    261 001010  016767  000000G 000000G          MOV     INPCNT,NAKCNT   ;NUMBER OF RECORDS IN RETRANSMIT
    262 001016  152767  000000G 000000G          BISB    #NAK,RESP       ;SEND A NAK
    263 001024  000167  177202                   JMP     GOASR
    264 001030                          SETB1:
```

```
                                                .SBTTL  VERTICAL FORMS CONTROL PROCESSING
                                        ;
001030  005002                          CLR     R2              ;CLEAR CARRIAGE CONTROL
001032  012405                          MOV     (R4)+,R5
001034  105767  000000G                TSTB    SWRBN           ;BINARY MODE
001040  001432                          BEQ     NOBINR          ;NO
001042  105767  000000G        BINARY: TSTB    SWRSL           ;SELECTIVE RECEIVE TOO?
001046  001533                          BEQ     LPGON           ;NO, JUST WRITE THE RECORD AS IS
001050  032764  001000  000002          BIT     #TRNREC,2(R4)   ;WAS THIS RECORD RECEIVED IN TRANSPARENCY
001056  001407                          BEQ     30$             ;NO
001060  022767  000000G 000000G 10$:    CMP     #CHANL,SWRSUP   ;PREVIOUS RECORD A PUNCH RECORD?
001066  001523                          BEQ     LPGON           ;YES, TREAT THIS ONE AS PUNCH DATA TOO
001070  004767  000374          20$:    JSR     PC,TRNSLT       ;NO, TRANSLATE THE DATA TO ASCII
001074  000414                          BR      NOBINR          ;CHECK FOR VFC AND HFC
001076  121527  000000G          30$:    CMPB    (R5),#BINESC    ;DOES THIS RECORD START WITH AN ESCAPE?
001102  001366                          BNE     10$             ;NO, TREAT LIKE THE LAST RECORD
001104  126527  000001  000000G         CMPB    1(R5),#BIN4     ;BINARY ESCAPE FOUR?
001112  001366                          BNE     20$             ;NO, TREAT AS A PRINT RECORD
                                                                ;OTHERWISE SELECT PUNCH CHANNEL & WRITE
                                        ;
001114  012767  000000G 000000G SELECT: MOV     #CHANL,SWRSUP   ;SELECT THE PUNCH CHANNEL
001122  005725                          TST     (R5)+           ;SCIP OVER ESC SEQUENCE
001124  000476                          BR      CHRCL           ;AND WRITE THE RECORD
                                        ;
001126  032764  001000  000002  NOBINR: BIT     #TRNREC,2(R4)   ;TRANSPARENT RECORD?
001134  001004                          BNE     10$             ;YES, IGNORE DEVICE SELECTION
001136  021527  032000G                 CMP     (R5),#64*400+ESC        ;ASCII ESCAPE FOUR?
001142  001010                          BNE     20$             ;NO, SEE IF PRINT SELECTION PRESENT
001144  000763                          BR      SELECT          ;YES, SELECT PUNCH CHANNEL
001146  105767  000000G          10$:    TSTB    SWRSL           ;SELECTIVE RECEIVE ACTIVE?
001152  001404                          BEQ     20$             ;NO, TREAT ALL ESCAPE SEQUENCES AS VFC
001154  022767  000000G 000000G         CMP     #CHANL,SWRSUP   ;YES, LAST RECORD A PUNCH RECORD?
001162  001465                          BEQ     LPGON           ;YES, TREAT THIS ONE A PUNCH THEN
001164  122527  000000G          20$:    CMPB    (R5)+,#ESC      ;ASCII ESCAPE?
001170  001062                          BNE     LPGON           ;NO, JUST WRITE LIKE THE LAST RECORD
001172  112502                          MOVB    (R5)+,R2        ;GET SECOND CHARACTER OF ESCAPE SEQUENCE
001174  105767  000000G                TSTB    SWRSL           ;SELECTIVE RECEIVE REQUESTED (/SL)
001200  001403                          BEQ     NOSLCT          ;NO, JUST ANALYZE PRINT CONTROL
001202  012767  000000G 000000G         MOV     #CHANL2,SWRSUP  ;YES, SELECT PRINT CHANNEL
                                                                ;AND ANALYZE FORMS CONTROL
                                        ;
001210                          NOSLCT:
```

```
                                                .SBTTL  HORIZONTAL FORMS CONTROL
                
001210  120227  000011                          CMPB    R2,#11          ;ESC HT SEQUENCE?
001214  001024                                  BNE     INPA            ;NO - DO DEFAULT VFC
001216  012702  000000G                         MOV     #LPSIZE,R2      ;YES - COPY RECORD INTO HFC STORAGE
001222  011403                                  MOV     (R4),R3         ;RECEIVE COUNT
001224  005743                                  TST     -(R3)           ;SUBTRACT TWO FOR ESC HT CHARS
001226  012700  000000G                         MOV     #HFCREC,R0      ;USE BUFFER TO STORE HFR
001232  112501                        HFRCOP: MOVB    (R5)+,R1        ;GET NEXT CHAR
001234  120127  000011                          CMPB    R1,#11          ;IS IT HT?
001240  001401                                  BEQ     HFRC1           ;YES - LEAVE AS IS
001242  005001                                  CLR     R1              ;NO - CLEAR IT
001244  110120                        HFRC1:  MOVB    R1,(R0)+        ;STORE CURRENT CHAR
001246  005302                                  DEC     R2              ;ROOM FOR MORE?
001250  001402                                  BEQ     HFRC2           ;NO - IGNORE THE REST OF THE RECORD
001252  005303                                  DEC     R3              ;MORE TO DO?
001254  001366                                  BNE     HFRCOP          ;YES - GO DO NEXT CHAR
001256  152767  000200  000000G HFRC2:  BISB    #200,HFCACT     ;SET 'HFC ACTIVE' FLAG
001264  000433                                  BR      LPFIN
                
```

```
                                                .SBTTL  CARRIAGE CONTROL
001266  042702  177600            INPA:   BIC     #177600,R2
001272  162702  000100                    SUB     #100,R2         ;IS CC AN A-H
001276  002411                            BLT     CHRCL           ;NO LESS THAN A
001300  020227  000010                    CMP     R2,#10          ;IN RANGE OF A-H
001304  003407                            BLE     CHRSL           ;YES
001306  162702  000012                    SUB     #12,R2          ;IS IT AN S OR T
001312  002403                            BLT     CHRCL           ;NO, CC IS H-R(ILLEGAL)
001314  020227  000013                    CMP     R2,#13          ;GREATER THAN T
001320  002401                            BLT     CHRSL           ;YES
001322  005002                    CHRCL:  CLR     R2
001324  010500                    CHRSL:  MOV     R5,R0           ,BUFFER ADDR IN R0
001326  011401                            MOV     (R4),R1         ;GET WORD COUNT
001330  062701  177776                    ADD     #-2,R1
001334  000402                            BR      LPGOC

001336  011401                    LPGON:  MOV     (R4),R1
001340  014400                            MOV     -(R4),R0                ;BUFFER ADDR IN R0
001342  004767  000000G           LPGOC:  JSR     PC,WRIT$                        ;WRITE
                                        .IF DF  RSX
001346  103002                            BCC     LPFIN           ;BRANCH IF NO ERROR
001350  004767  000000G                    JSR     PC,CLOSER       ;OTHERWISE CLOSE THE FILES
                                        .ENDC
001354  105267  000000G           LPFIN:  INCB    CMBFAV          ;
001360  105267  000000G                   INCB    INPCNT          ;INCREMENT COUTER
001364  106367  000000G                   ASLB    ACKPEN          ;HAVE WE SEEN AN ETX ETB
001370  103005                            BCC     SETB6           ;NO
001372  105067  000000G                   CLRB    INPCNT          ;CLEAR INPUT COUNT
001376  152767  000000G 000000G           BISB    #ACK,RESP       ;SEND AN ACK
001404  004767  000004            SETB6:  JSR     PC,ASROUT       ;ASSIGN BUFFER
001410  000167  000000G                   JMP     GETENT
```

```
                                                .SBTTL  ASSIGN BUFFER TO BYSYNC TAP

001414  116700  000000G         ASROUT.: MOVB    ASSGNX,R0              ;FIND CURRENT ASSIGN SLOT...
001420  016067  000000G 000002G          MOV     CMBLST(R0),BISPAR+2    ;..AND MOVE BUFFER ADDRESS TO ARGLIST
001426                                   ASRBUF  BISPAR                         ;ASSIGN TO BISYNC TAP
001436  142767  000140  000001G          BICB    #140,BISPAR+1          ;RESET THE OP-CODE
001444  105367  000000G                  DECB    CMBFAV
001450  005720                          TST     (R0)+                  ;UPDATE SLOT INDEX
001452  020027  000000G                  CMP     R0,#CMBMAX             ;NEED TO RESET IT?
001456  103401                          BLO     ASRZ
001460  005000                          CLR     R0
001462  110067  000000G         ASRZ:    MOVB    R0,ASSGNX              ;SAVE UPDATED INDEX
001466  000207                          RTS     PC
```

```
RECEIVE MACRO M1104  06-DEC-77 08:28  PAGE 12
TRANSLATION ROUTINE FOR PRINT DATA (/BI/SL)


                                        .SBTTL  TRANSLATION ROUTINE FOR PRINT DATA (/BI/SL)
                                ;
                                ;+
                                ; INPUTS:
                                ;       R4=POINTER TO BYTE COUNT
                                ;       R5=POINTER TO DATA
                                ;-
001470  011401          TRNSLT: MOV     (R4),R1         ;GET THE BYTE COUNT
001472  010546                  MOV     R5,-(SP)        ;SAVE DATA POINTER
001474  111500          10$:    MOVB    (R5),R0         ;GET THE NEXT CHARACTER
001476  042700  177400          BIC     #177400,R0      ;CLEAR THE HIGH BYTE
001502  116025  000000G         MOVB    ETA(R0),(R5)+   ;GET ITS EQUIVALENT (TABLE DEFINED IN TAP)
001506  005301                  DEC     R1              ;DECREMENT CHARCTER COUNT
001510  001371                  BNE     10$             ;MORE TO GO
001512  012605                  MOV     (SP)+,R5        ;RESTORE DATA POINTER
001514  000207                  RTS     PC              ;ALL     DONE
```

```
RECEIVE MACRO M1104  06-DEC-77 08:28  PAGE 13
MOVE DATA TO FNB OR COMMON SWITCH AREA

                                                .SBTTL  MOVE DATA TO FNB OR COMMON SWITCH AREA
                                        ;
                                        ; INPUTS:
                                        ;       R0=DESTINATION
                                        ;       R1=# OF WORDS TO MOVE
                                        ;       R2=SOURCE
                                        ;
001516                                  XFRFNB::
001516  012220                                  MOV     (R2)+,(R0)+             ;MOVE ONE WORD
001520  005301                                  DEC     R1                      ;DECREMENT THE WORD COUNT
001522  001375                                  BNE     XFRFNB                  ;MORE TO GO
001524  000207                                  RTS     PC                      ;ALL DONE
```

```
RECEIVE MACRO M1104  06-DEC-77 08:28  PAGE 14
CONSTANT DATA AREA

                                                .SBTTL  CONSTANT DATA AREA
                                        ;
001526  070513                          QRECE:  .RAD50  'REC'
                                        ;
000001                                          .END
```

```
RECEIVE MACRO M1104  06-DEC-77 08:28  PAGE 14-1
SYMBOL TABLE

ACK    = ****** GX        ENTER = ****** GX        INPRA   000514R     002 RCVSWT= ****** GX        SWRSL  = ****** GX
ACKPEN= ****** GX         ENTRY = ****** GX        INPR2   000552R     002 RECS1 = ****** GX        SWRSUP= ****** GX
ASROUT  001414RG    002 ESC   = ****** GX        IOERR = ****** GX        RESP  = ****** GX        S.FNBW= ****** GX
ASRZ    001462R     002 ETA   = ****** GX        LPFIN   001354R     002 RSX   = 000001           TAGE1   000017R     003
ASSGNX= ****** GX         ETXFLG= ****** GX        LPGOC   001342R     002 RSX11M= 000001           TAGE2   000034R     003
BINARY  001042R     002 FDBOUO= ****** GX        LPGON   001336R     002 RTYP  = ****** GX        TAGE3   000046R     003
BINESC= ****** GX         FD.CR = ****** GX        LPSIZE= ****** GX        RVAL  = ****** GX        TAGE4   000063R     003
BIN4   = ****** GX         FLAGS = ****** GX        MODE  = ****** GX        RVIPEN= ****** GX        TAGNUM= 000004
BISPAR= ****** GX         F.FNB = ****** GX        NAK   = ****** GX        RV1ERR  000526R     002 TAG1  000000R     003
CHANL  = ****** GX         GETENT= ****** GX        NAKCNT= ****** GX        R.VAR = ****** GX        TAG2  000017R     003
CHANL2= ****** GX         GOASR   000232R     002 NOBUFR  001126R     002 SELECT  001114R     002 TAG3  000034R     003
CHRCL   001322R     002 GODSC   000204R     002 NOKE    000210R     002 SETBW   000564R     002 TAG4  000046R     003
CHRSL   001324R     002 HFCACT= ****** GX        NOSLCT  001210R     002 SETBO   000644R     002 TRNREC= 001000
CLOSER= ****** GX         HFCREC= ****** GX        NRCVSF= 000000           SETB1   001030R     002 TRNSLT  001470R     002
CMBFAV= ****** GX         HFRCOP  001232R     002 NTRNSF= 000002           SETB2   000712R     002 TSTPFL  000534R     002
CMBLST= ****** GX         HFRC1   001244R     002 OFFHAN= ****** GX        SETB3   000774R     002 TURNAR= ****** GX
CMBMAX= ****** GX         HFRC2   001256R     002 ONLINE= ****** GX        SETB4   000740R     002 TXPEN = ****** GX
COLP1  = ****** GX         INCP2   000246R     002 OPNFLG= ****** GX        SETB5   000600R     002 TYPMSG= ****** GX
COLP2  = ****** GX         INCRCV= ****** GX        PROMPT= ****** GX        SETB6   001404R     002 UBFCNT= ****** GX
COLTT  = ****** GX         INITF = ****** GX        QMODE = ****** GX        SLAVC1  000316R     002 UBFHAN= ****** GX
DFCNT   000440R     002 INPA    001266R     002 QRECE   001526R     002 SNDEOT= ****** GX        UOVR  = 000020
DFLT    000362R     002 INPCNT= ****** GX        RATT  = ****** GX        STKSET= ****** GX        WRITS = ****** GX
DFNLP  = ****** GX         INPCPL  000000RG    002 RCHN  = ****** GX        SWRBN = ****** GX        XFRFNB  001516RG    002
EMTPUT= ****** GX         INPEXT  000242R     002 RCVBLK= ****** GX

. ABS.  000000    000
        000000    001
MAIN    001530    002
MESAG2  000063    003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:   1465 WORDS  ( 6 PAGES)
DYNAMIC MEMORY:   2810 WORDS  ( 10 PAGES)
ELAPSED TIME:   00:00:32
 256,020 RECV;27, 256,030 RECV/CRF/-SP=UAPRXM,CTXMAC,RECV
```

RECV CREATED BY MACRO ON 6-DEC-77 AT 08:28 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES					
ACK	= ******	GX	5-121	9-360				
ACKPEN	= ******	GX	*6-217	*6-223	*6-233	*6-254	*9-357	
ASROUT	001414	RG	5-123	6-222	9-361	#11-370		
ASRZ	001462	R	11-377	#11-379				
ASSGNX	= ******	GX	11-370	*11-379				
BINARY	001042	R	#7-272					
BINESC	= ******	GX	7-280					
BIN4	= ******	GX	7-282					
BISPAR	= ******	GX	*11-371	11-372	*11-373			
CHANL	= ******	GX	7-276	7-286	7-297			
CHANL2	= ******	GX	7-304					
CHRCL	001322	R	7-288	9-335	9-339	#9-342		
CHRSL	001324	R	9-337	9-341	#9-343			
CLOSER	= ******	GX	5-102	9-353				
CMBFAV	= ******	GX	*5-81	*5-122	*6-221	*9-355	*11-374	
CMBLST	= ******	GX	11-371					
CMBMAX	= ******	GX	11-376					
COLP1	= ******	GX	*6-186					
COLP2	= ******	GX	*6-187					
COLTT	= ******	GX	6-186	6-187				
DFCNT	000440	R	6-153	#6-176				
DFLT	000362	R	6-146	#6-154				
DFNLP	= ******	GX	6-167					
EMTPUT	= ******	GX	11-372					
ENTER	= ******	GX	6-192					
ENTRY	= ******	GX	5-77	5-78				
ESC	= ******	GX	7-292	7-299				
ETA	= ******	GX	12-393					
ETXFLG	= ******	GX	*5-85	*6-210	*6-235	*6-249		
FDBOUO	= ******	GX	6-144					
FD.CR	= ******	GX	6-175					
FLAGS	= ******	GX	5-87	5-126				
F.FNB	= ******	GX	6-144					
GETENT	= ******	GX	5-124	9-362				
GOASR	000232	R	5-76	#5-122	5-127	5-135	6-238	6-256	6-263
GODSC	000204	R	#5-116	5-133				
HFCACT	= ******	GX	*8-328					
HFCREC	= ******	GX	8-318					
HFRCOP	001232	R	#8-319	8-327				
HFRC1	001244	R	8-321	#8-323				
HFRC2	001256	R	8-325	#8-328				
INCP2	000246	R	5-80	#5-126				
INCRCV	= ******	GX	*5-95	*6-190				
INITF	= ******	GX	5-128					
INPA	001266	R	8-314	#9-333				
INPCNT	= ******	GX	*6-220	*6-226	6-236	*6-251	*6-259	6-261	*9-356	*9-359
INPCPL	000000	RG	#5-75					
INPEXT	000242	R	#5-124	6-224				
INPRA	000514	R	5-129	6-193	#6-198			
INPR2	000552	R	6-209	#6-212				
IOERR	= ******	GX	*6-195	6-218	*6-225			
LPFIN	001354	R	8-329	9-352	#9-355			

RECV CREATED BY MACRO ON 6-DEC-77 AT 08:28 PAGE 2
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES							
LPGOC	001342 R		9-346	#9-350						
LPGON	001336 R		7-273	7-277	7-298	7-300	#9-348			
LPSIZE	= ******	GX	8-315							
MODE	= ******	GX	5-91	*5-119	5-134					
NAK	= ******	GX	6-237	6-262						
NAKCNT	= ******	GX	6-229	*6-232	*6-236	*6-252	*6-253	*6-258	6-259	*6-261
NOBINR	001126 R		7-271	7-279	#7-290					
NOKE	000210 R		#5-119	5-137						
NOSLCT	001210 R		7-303	#7-307						
NRCVSF	= 000000		#1-5							
NTRNSF	= 000002		#1-4							
OFFHAN	= ******	GX	5-116							
ONLINE	= ******	GX	5-75							
OPNFLG	= ******	GX	*5-93	*5-136	*6-199					
PROMPT	= ******	GX	5-106	6-194						
QMODE	= ******	GX	*5-120							
QRECE	001526 R		5-120	#14-416						
RATT	= ******	GX	*6-175							
RCHN	= ******	GX	*6-172							
RCVBLK	= ******	GX	6-142							
RCVSWT	= ******	GX	6-149							
RECS1	= ******	GX	*6-188							
RESP	= ******	GX	*5-121	*6-237	*6-262	*9-360				
RSX	= 000001		#1-2	6-166	6-185	9-351				
RSX11M	= 000001		#1-3							
RTYP	= ******	GX	*6-174							
RT11	= ******		6-156	6-178						
RVAL	= ******	GX	*6-173							
RVIPEN	= ******	GX	*5-108							
RV1ERR	000526 R		5-131	#6-204						
R.VAR	= ******	GX	6-174							
SELECT	001114 R		#7-286	7-294						
SETBW	000564 R		6-205	6-211	6-213	#6-215				
SETBO	000644 R		6-219	#6-229						
SETB1	001030 R		6-241	6-246	#6-264					
SETB2	000712 R		6-231	#6-240						
SETB3	000774 R		6-255	#6-258						
SETB4	000740 R		6-230	#6-249						
SETB5	000600 R		6-216	#6-218						
SETB6	001404 R		9-358	#9-361						
SLAVC1	000316 R		#6-142	6-200						
SNDEOT	= ******	GX	5-90	6-227						
STKSET	= ******	GX	6-248							
SWRBN	= ******	GX	6-150	*6-170	7-270					
SWRSL	= ******	GX	*6-171	7-272	7-295	7-302				
SWRSUP	= ******	GX	7-276	*7-286	7-297	*7-304				
S.FNBW	= ******	GX	6-143							
TAGE1	000017 R		5-99	#5-99						
TAGE2	000034 R		5-101	#5-101						
TAGE3	000046 R		5-115	#5-115						
TAGE4	000063 R		6-194	#6-194						
TAGNUM	= 000004		#1-8	*5-99	#5-99	5-99	5-101	#5-101	5-101	5-115	#5-115

SYMBOL	VALUE		REFERENCES				
			5-115	6-194	#6-194	6-194	
TAG1	000000	R	5-99	5-99	#5-99		
TAG2	000017	R	5-101	5-101	#5-101		
TAG3	000034	R	5-115	5-115	#5-115		
TAG4	000046	R	6-194	6-194	#6-194		
TRNREC	= 001000		#4-67	7-274	7-290		
TRNSLT	001470	R	7-278	#12-389			
TSTPFL	000534	R	6-202	#6-208			
TURNAR	= ******	GX	5-114				
TXPEN	= ******	GX	*5-110				
TYPMSG	= ******	GX	5-99	5-101	5-115	6-194	
UBFCNT	= ******	GX	5-103	*6-245			
UBFHAN	= ******	GX	5-105				
UOVR	= 000020		#4-68	6-242			
WRITS	= ******	GX	9-350				
YERFNB	001516	RG	6-148	6-151	6-168	#13-406	13-409

MACRO CROSS REFERENCE CREF

MACRO NAME	REFERENCES				
ASRBUF	#3-28	11-372			
CLOSE$	#4-64				
ERROR	#1-9	6-194			
ERRORK	#1-16	5-99	5-101	5-115	6-194
ERRORM	#1-32				
ERRORP	#1-47				
FATAL	#1-53				
GETS	#3-43				
LINIT	#3-23				
PUTD	#3-52				
PUTMC	#3-38				
PUTTC	#3-33				
PUTTM	#3-47				

```
                                ;
                                ;
                                ;CONFIGURATION FILE FOR RSX11M 2780
                                ;V00-01 JLH 2/19/74
000004                          SWB=    4       ;RSX11M CORESIDENCY
000001                          DPPRES= 1       ;DP11 PRESEN
000001                          BIPRES= 1       ;BISYNC TAP PRESENT
000000                          CKW11=  0       ;NO CLOCK INIT
070000                          C.SMAX= 70000   ;CONFIGURE FOR 28K
000000                          EMTVEC= 0       ;JSR PC,CDEMT ENTRY
000020                          C.USFY=20       ;20 UIB ENTRIES
000000                          SLOCK=0 ;NO SOFTWARE LOCK
000001                          TLOCK=1         ;TIMER LOCK
000030                          CUTB=   30      ;24 TIMER QUEUE ENTRIES
000000                          CUTNPS= 0       ;NO TIMER SCHEDULIN
000001                          C.USNL= 1       ;MAX LINE NTMBER = 1
000400                          C.USRT= 400     ;256 WORD LINE TABLE SAPCE
000001                          TAPMAX= 1       ;MAX TAP UTILIZED
000003                          ISRMAX= 3       ;MAX ISR UTILIZED
000000                          UIBSRH= 0       ;DO NOT ASSEMBLE FEATURE TO ALLOW UIB SEARCH
                                                ;FOR A PARTICULAR LINE NUMBER
000001                          RSX11=  1       ;***SPECIAL SWITCH TO CUT CODE
000001                          FAKELU= 1       ;USE FAKE LOADER/UNLOADER
                                ;PREFIX FILE TO DEFINE RSX-11M SYSTEM SYMBOLS VIA MACRO CALLS
                                ;
000000                          PLF=    0       ;TAKE OUT LEVEL F
                                ;DEFINE TASK CONTROL BLOCK OFFSETS
                                .MCALL TCBDF$
000000                          TCBDF$
                                ;
```

```
                                ;PREFIX FILE FOR RSX11M ASSEMBLER AND ANY OTHER ASSEMBLER
                                ;WHICH DOES NOT RECOGNIZE OPCODE MNEMONICS EXCEPT WHEN THEY
                                ;APPEAR IN THE OPCODE FIELD
000000                          HALT=0
000240                          NOP=240
000241                          CLC=241
000242                          CLV=242
000244                          CLZ=244
000250                          CLN=250
000257                          CCC=257
000261                          SEC=261
000262                          SEV=262
000264                          SEZ=264
000270                          SEN=270
000277                          SCC=277
```

```
                        ;SCIP-11
                        ;SYSTEM CONTROL AND INTERFACE PROGRAM
                        ;COMTEX-11 SCIP
                        ;COPYRIGHT (C) 1970,1971,1972,1973,1974,1975,1976,1977
                        ;DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                        ;
                        ;THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY
                        ;ON A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                        ;THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                        ;SOFTWARE, OR ANY COPIES THEREOF, MAY NOT BE PROVIDED OR
                        ;OTHERWISE MADE AVBAILABLE TO ANY OTHER PERSON EXCEPT
                        ;FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO THESE
                        ;LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE SOFTWARE
                        ;SHALL AT ALL TIMES REMAIN IN DEC.
                        ;
                        ;THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                        ;WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                        ;MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                        ;
                        ;DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                        ;OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                        ;DEC.
                        ;
                        ;V03-01 JLH 6/12/73
                        ;V03-02 JLH 7/10/73
                        ;V03-03 JLH 10/15/73
                        ;V03-04 JLH 1/29/74
                        ;V03-05 JLH 2/5/74
                        ;V03-06 JLH 2/12/74
                        ;V03-07 JLH 2/18/74
                        ;V03-08 JLH 2/19/74
                        ;V03-09 JLH 4/22/74
                        ;V03-10 JLH 7/1/74
                        ;V03-11 RHR 7/17/74
                        ;V03-12 RHR 8/7/74
                        ;V03-13 RHR 8/9/74
                        ;V03-14 RHR 10/1/74
                        ;V03-15 JLH 11/18/74
                        ;V03-16 JLH 1/15/75
                        ;V03-17 RHR 2/15/75
                        ;V03-18 JLH 3/12/75
                        ;V03-19 SJP 9/15/75
                        ;V03-20 CEP 2/77
                        ;V03-21 CEP 6/77
                        ;V03-22 CEP 8/77
                        ;
                        ;THE DIFFERENCES BETWEEN V03 AND V02 ARE  THE FOLLOWING:
                        ;1) REARRANGEMENT OF CODE TO ENABLE EASIER ACCESS TO
                        ;   GIVEN ROUTINES.
                        ;2) DELETION OF C.ST* LABELS IN THE PRIORITY SCHEDULER
                        ;   DEFINITIONS. MODIFICATION OF SEQUENCE
                        ;       JMP @(PC)+
                        ;C.ST*: .WORD C.SEA*
                        ;TO
                        ;       JMP C.SEA*
                        ;3) ADDITION OF RESTART CAPABILITY. EMT 103 OP CODE 1 CLEARS
                        ;   C.STRT AND EMT 103 OPCODE 0 (LINIT) INITIALIZES ALL
```

```
                        ;    NECCESSARY TABLES, POINTERS, QUEUES, ETC ON FIRST
                        ;    ENTRY.
                        ;4) REMOVAL OF EMBEDDED VARIABLES.
                        ;5) MODIFICATION OF THE EMT SERVICE ROUTINE TO CHECK PARAMETER
                        ;    POINTER FOR LEGAL WORD ADDRESS BEFORE CALLING
                        ;    EMT SERVICE ROUTINES.
                        ;6) MODIFICATION OF EMT SERVICE ROUTINES TO OPERATE AS
                        ;    SUBROUTINES.
                        ;7) MODIFICATION OF SIB UNLOADER TO CONDITIONALLY DELETE
                        ;    SIB LOADER/UNLOADER IF NUMBER OF ENTRIES IN BUFFER IS
                        ;    SET TO ZERO.  THIS ALLOWS CONVENIENT USE OF
                        ;    THE PL* CODE WHEN THE SIB FOR  THAT LEVEL IS NOT
                        ;    DESIRED.
                        ;8) MODIFICATION OF THE TIMER CODE TO DELETE THE PROGRAM LEVEL
                        ;    SCHEDULING ROUTINE IF CUTNPS=0 AND DELETION OF
                        ;    THE TIMER QUEUE SERVICE ROUTINES IF CUTB=0
                        ;9) MODIFICATION OF THE TIMER QUEUE SERVICE ROUTINE TO
                        ;    RETURN THE FREE LINK TO THE VACANCY CHAIN PRIOR TO
                        ;    SERVING THE CURRENT ENTRY, I.E.  A SYSTEM CAN NOW RUN
                        ;    WITH ONE LINK AND HAVE EACH ROUTINE CAUSE ANOTHER TIMER
                        ;    ENTRY TO BE QUEUED
                        ;A) THE INITIALIZATION ROUTINE WILL NORMALLY CHECK TO DETERMINE
                        ;    WHICH CLOCK IS PRESENT.  THE RULES ARE THE FOLLOWING:
                        ;    - IF THE USER SPECIFIES A CLOCK, THAT CLOCK WILL BE UTILIZED.
                        ;    - IF NO CLOCK IS SPECIFIED, THE LINE CLOCK WILL BE UTILIZED.
                        ;    - OTHERWISE THE PROGRAMMABLE CLOCK WILL BE UTILIZED.
                        ;B) THE INITIALIZATION ROUTINE WILL NORMALLY CHECK TO DETERMINE
                        ;    MAXIMUM CORE PRESENT IN THE SYSTEM.  THE RULES ARE THE
                        ;    FOLLOWING:
                        ;    - IF THE USER SPECIFIES A MEMORY SIZE, THAT SIZE WILL BE USED
                        ;    - OTHERWISE, THE INIT RTN WILL FIND FIRST 1K ADDRESS NOT PRESENT
                        ;      AND PLACE THAT ADDRESS IN C.SMXM.
                        ;C) ADDITION OF DIFFERENT EMT VECTOR ADDRESS SUPPORT V03-03
                        ;D) MODIFICATION OF PS SAVES/RESTORES TO UTILIZE BYTE INST.
                        ;    WHERE APPLICABLE.  NOT APPLICABLE IN SLOCK CODE.
                        ;E) FIX OF BUG IN SIB BUFFER LOADER.  V03-03
                        ;F) V03-04 FIX OF PROBLEM IN DOS REGISTER SAVE ROUTINE BY DELETION
                        ;    OF COMTEX-11 CALL TO DOS ROUTINE FOR REGISTER SAVE.
                        ;G) V03-04 FIX OF PROBLEM IN BUFFER POOL INITIALIZATION ROUTINE
                        ;          CAUSED BY ERRONEOUS USER CALL, I.E.  WITH ODD
                        ;          BEGINNING OR ENDING ADDRESS.  RETURNS WITH ERROR.
                        ;H) V03-05 ADDITION OF EMTVEC=0 CODE IMPLYING CALL VIA JSR PC,CDEMT
                        ;I) V03-05 ADDITION OF C.USFY=0 IMPLYING UIB LOADER CALL TO
                        ;          TO USER PROGRAM THROUGH TABLE C.SFUJ.
                        ;J) V03-05 ADDITION OF SWB=2 CODE FOR RSTS/E
                        ;K) V03-05 ADDITION OF C.USRT=0 FOR ZERO LINE TABLE SAPCE
                        ;L) V03-06 CHANGE OF DEFAULTS FOR TAP AND ISR PRESENCE
                        ;          AND PROGRAM LEVEL PRESENCE TO NOT PRESENT FOR EASE
                        ;          OF CONFIGURATION.
                        ;M) V03-06 ADDITION OF TAPMAX AND ISRMAX TO ALLOW DELETION OF
                        ;          UNUSED TABLE SPACE FOR NOT INCLUDED TAPS AND ISRS.
                        ;N)  V03-07 ADDITION OF SWB=3 CODE FOR RSX-11 D USAGE
                        ;O) V03-08 ALL SYSTEM CALLS INCLUDING MATRIX SWITCH, TIMER QUEUE
                        ;          AND PRIORITY ENTRANCE EXIT CALLS ARE VIA JSR PC,XXXX
                        ;          C.SARR, CTRET, AND C.SEC* ARE DEFINED TO BE RTS PC
                        ;P) V03-09 CORRECTIONS TO GLOBAL DEFINITIONS IN PRIORITY
                        ;          SCHEDULER, MODIFICATION OF RSTS/E VERSION TO
```

```
                                ;           UTILIZE BIG BUFFERS FOR TAP LINE TABLE SPACE
                                ;           AND MODIFICATION OF THE RSX11D VERSION TO
                                ;           UTILIZE EVENT FLAGS FOR SIB UNLOADING.
                                ;Q) V03-10 MODIFICATIONS TO RUN WITH RSX11M
                                ;R) V03-11 CORRECTIONS TO CONDITIONAL CODE DEPENDENT ON C.SMAX AND CKW11
                                ;           DEFINE "SERCH" EQUAL TO THE PRODUCT AND CONDITIONALIZE
                                ;           ON IT.
                                ;S) V03-12 ADD CODE FOR RSX-11D/M TO ZERO ALL TIMER ENTRIES FOR A
                                ;           RESTART AND THEN TO DO A CANCEL MARK TIME DIRECTIVE TO
                                ;           PREVENT A TIMER GOING OFF AFTER THE QUEUE HAS BE_N
                                ;           REINITIALIZED
                                ;T) V03-13 CONDITIONALIZE UIB UNLOADER ROUTINE WHICH SEARCHES
                                ;           FOR A PARTICULAR LINE NUMBER.  "UIBSRH"=0 INDICATED
                                ;           THIS FEATURE IS NOT DESIRED. UIBSRH NOT EQUAL 0 INDICATES
                                ;           THAT IT IS.
                                ;U) V03-14 DELETE A TEMPORARY BRANCH TO SELF IN CTIME
                                ;V)  V03-15 CORRECTION OF BUFFER POOL MANAGEMENT ROUTINE TO
                                ;    CORRECT ERROR IN GET BUFFER ROUTINE IN 3 WORD BUFFER INFO
                                ;    BLOCK POSSIBLY OVERWRING CONTENTS OF A REAL DATA BUFFER
                                ;W) V03-16 CORRECT CONDITIONALIZATION
                                ;X) V03-17 MOVE GLOBL DEFINITION OF C.START UNDER CONDITIONALIZATION
                                ;Y) V03-18 ADDITION OF BR LEVEL CODE CONDITIONALIZATION TO ALLOW
                                ;   DELETION OF EXTRANEOUS PROGRAM LEVEL CODE.
                                ;   ADDITION OF TLOCK CONDITIONAL TO ALLOW DELETION OF ANY LOCK AROUND
                                ;   TIMER QUEUE SERVICE CODE IF THAT CODE IS ONLY CALLED BY USER LEVEL.
                                ;   MOVING OF THE TIMER PRIORITY LEVEL TO BELOW THAT OF BR4
                                ;Z) V03-19 MODIFICATIONS TO RUN WITH RT-11
                                ;AA) V03-20  MODIFICATIONS TO CUT CODE FOR RSTS V&B
                                ;      DONE BY USING 'RSTS' CONDITIONAL SWITCH
                                ;BB) V03-21  MODIFICATIONS TO ALLOW PROGRAM TO RUN ON LSI11
                                ;           DONE USING 'LSI' CONDITIONAL SWITCH
                                ;CC) V03-22  MODIFICATIONS TO CUT CODE
                                ;
                                ;THE DIFFERENCES BETWEEN V02 AND V001A ARE THE FOLLOWING:
                                ;1) ADDITION OF DYNAMIC CORE BUFFERING WITH EMT SERVICE
                                ;   TO ALLOCATE AND DEALLOCATE CORE SPACE AND TO OBTAIN
                                ;   AND RELEASE DYNAMIC BUFFERS. SUBROUTINE LINKAGE IS
                                ;   AVAILABLE TO PERFORM THE LATTER TASKS TO ALLOW TAP/ISR'S
                                ;   TO UTILIZE THE BUFFER POOL.
                                ;2) THE ADDITION OF FURTHER CHECKS OF VALIDITY ON THE MATRIX
                                ;   SWITCH CONTENTS.
                                ;3) MODIFICATION OF THE SIB UNLOADER TO VECTOR ON ENTRY TYPE
                                ;   CODE AND THE TAP TYPE CODE. THIS PREVENTS THE NECCESSITY OF
                                ;   REASSEMBLING THE ISR TABLES FOR DIFFERENT ASSIGNMENTS OF
                                ;   THE TAPS.
                                ;4) FURTHER VALIDITY CHECKS OF THE SIB UNLOADER VECTOR TABLE.
                                ;5) DELETION OF ENTRIES IN THE TIMER QUEUE.
                                ;6) MODIFICATION OF MATRIX SWITCH VECTORING TO ALLOW DELTION OF
                                ;   UNUSED TABLES IN SCIP WITH NO ADVERSE EFFECT ON OPERATION.
                                ;7) ADDITION OF DOS/COMTEX SUPPORT CHANGES
                                ;8) ADDITION OF SOFTWARE LOCK SOFTWARE PRIORITY LEVEL, ASSUMED TO
                                ;   OPERATE AT A LEVEL HIGHER THAN ANY OTHER SOFTWARE TASK TO
                                ;   ALLOW LOCKING OF LONG EXECUTION TASKS - SOFTWARE REQUESTED
                                ;   WITHOUT LOCKING OUT INTERRUPT DEVICES.
                                ;9) ADDITION OF CODE IN THE EMT CALL HANDLERS TO DETECT
                                ;   ADDITIONAL FORMAT ERRORS. THE CONDITION CODE SETTINGS
                                ;   RETURNED TO THE USER PROGRAM HAVE BEEN EXTENDED AND
```

```
                        ;    STANDARDIZED.
                        ;
                        ;THE DIFFERENCES BETWEEN SCIP V001A AND SCIP V000A ARE:
                        ;1) A DEFINITION OF AVAILABLE CORE SIZE (C.SMAX) USED IN A
                        ;    GLOBAL LOCATION (C.SMXM) SO THAT THE ISR'S AND TAP'S CAN
                        ;    DETECT OVERSIZED BUFFERS.  (IT IS NOT USED BY THE SCIP.)
                        ;2) THE ADDITION OF CODE FOR THE KILLIO EMT COMMAND.
                        ;3) THE REPLACEMENT OF THE PUT POLL TABLE IN THE TERMINAL
                        ;    TYPE CODE TABLES BY A MODEM CONTROL TABLE.
                        ;4) A TEST FOR LINE NUMBER VALIDITY IN AN EMT 100 CALL.
```

```
                                .TITLE SCP322 SCIP VERSION 3-22
                        ;DEFINITIONS OF REGISTERS AND CONDITION CODE OPERATORS
                        ;
177776                  C.STAT= 177776  ;PROCESSOR STATUS REGISTER
000000                  R0=     %0
000001                  R1=     %1
000002                  R2=     %2
000003                  R3=     %3
000004                  R4=     %4
000005                  R5=     %5
000006                  SP=     %6
000007                  PC=     %7
                        ;
                        ;INSTRUCTION DEFINITIONS
000240                  NOP=    240     ;NO OPERATION INSTRUCTION
000277                  SETALL= SEZ!SEC!SEV!SEN ;SET ALL CONDITION CODES
000275                  SETNZC= SEN!SEZ!SEC     ;SET N,Z,C CONDITION CODES
000274                  SETNZ=  SEN!SEZ         ;SET N,Z CONDITION CODES
000257                  CLALL=  CLN!CLC!CLV!CLZ ;CLEAR ALL CONDITION CODES
000254                  CLZN=   CLZ!CLN         ;CLEAR Z,N CONDITION CODES
000243                  CLVC=   CLV!CLC         ;CLEAR V,C CONDITION CODES
                        ;
                        ;BIT DEFINITIONS
000001                  BIT00=  1
000002                  BIT01=  2
000004                  BIT02=  4
000010                  BIT03=  10
000020                  BIT04=  20
000040                  BIT05=  40
000100                  BIT06=  100
000200                  BIT07=  200
000400                  BIT08=  400
001000                  BIT09=  1000
002000                  BIT10=  2000
004000                  BIT11=  4000
010000                  BIT12=  10000
020000                  BIT13=  20000
040000                  BIT14=  40000
100000                  BIT15=  100000
```

```
                                ;SCIP SYSTEM GLOBALS
                                .GLOBL C.SPS,C.SFY,C.SARR,C.SARP,C.SATP
                                .GLOBL C.SREG,C.SRES
                                .GLOBL C.STAB,CTRET
                                .GLOBL C.SECA,C.SECB,C.SECD,C.SECE,C.SECF,C.SECG,C.SECH


                                .IF Z SWB-5             ;RT-11
                                .MCALL ..V2...MRKT,.CMKT
                                ..V2..
                                .ENDC

                                ;DEFINITION OF PRIORITY LEVEL OF PRIORITY SCHEDULER
                                .IF NDF C.USP
                                C.USP=300       ;OCTAL PS SETTING
                                .ENDC

                                ;EMTVEC - DEFINITION OF EMT VECTOR ADDRESS
                                ;       =0 IF EMT SERVICE ENTERED VIA JSR PC,CDEMT
                                ;       ++ IF EMT SERVICE ENTERED VIA PROCESSOR TRAP AT "EMTVEC"
                                .IF NDF EMTVEC
                                EMTVEC= 30      ;DEFAULT VECTOR ADDRESS = 30
                                .ENDC
                                ;
                                ;SWB    - DEFINITION OF SYSTEM CO-RESIDENCY
                                ;       =2 RSTS/E CORESIDENCY "RSTS/E DRIVER"
                                ;       =3 RSX-11D CORESIDENCY
                                ;       =4 RSX-11M CORESIDENCY
                                ;       =5 RT-11  CORESIDENCY
000000                          SWBC=<SWB-3>*<SWB-4>*<SWB-5>    ;0 IF RSX11D OR RSX11M OR RT-11
                                .IF Z SWB-2
                                .GLOBL L3QUE
                                .ENDC
                                ;
                                .IF Z SWB-4     ;RSX11M PRESENCE FLAG
                                .GLOBL T.EFLG,$TKTCB,$RQSCH,$SIGFL
000000                          .PSECT ISR,CON
000000                          .PSECT XXX
                                .IF NDF UIBEF
                                UIBEF=1         ;UIB ENTRY PRESENT EVENT FLAG
                                .ENDC
                                .IF NDF SIBEF
                                SIBEF=40        ;SIB ENTRY PRESENT EVENT FLAG
                                .ENDC
                                .ENDC
                                .IF NZ SWB-3    ;IF NOT RSX11D
000000                          OFFSET=0
                                .ENDC
                                .IF Z SWB-3     ;RSX11D PRESENCE FLAG
                                .GLOBL .CRTSK,A.EF,EV.SE,.SERFG
                                .GLOBL R.TD,..PENP,A.AS,A.TS,TS.AST,YF.IC
                                .GLOBL Y.AE,Y.TT,..NADD
                                .PSECT ISR,CON
                                .PSECT XXX
                                OFFSET= 60000
                                .IF NDF UIBEF
                                UIBEF=  1       ;UIB ENTRY PRESENT EVENT FLAG
```

```
                        .ENDC
                        .IF NDF SIBEF
                        SIBEF=  40                      ;SIB ENTRY PRESENT EVENT FLAG
                        .ENDC
                        .ENDC
                        ;
```

```
                                        .IF DF RSX11
000000                                  .PSECT ISR
                                        .ENDC
                                ;DEVICE TYPE TABLES
                                ;MATRIX SWITCH ON DEVICE TYPE - FUNCTION CODE
                                ;C.SAR:  CONTAINS THE ADDRESS OF THE FUNCTION TABLES FOR
                                ;ALL ISRS INCLUDED IN A COMPLETE COMTEX SYSTEM.
                                ;
000000  000010                  C.SAR:  .WORD C.SARL     ;NUMBER OF ENTRIES IN TABLE *2
000002  000000                          .WORD CAFTBL     ;POINTER TO ISR FUNCTION TABLE FOR DC11
        000003                  X=ISRMAX
                                .IF NZ X
000004  000000                          .WORD CMFTBL     ;1= DM11 ISR
        000002                  X=X-1
                                .ENDC
                                .IF NZ X
000006  000000                          .WORD CCFTBL     ;2=KL11 ISR
        000001                  X=X-1
                                .ENDC
                                .IF NZ X
000010  000012'                         .WORD CPFTBL     ;3= DP11 ISR
        000000                  X=X-1
                                .ENDC
        000010                  C.SARL= .-C.SAR-2
                                ;
                                ;EACH ISR SHOULD THEN HAVE A TABLE OF THE FOLLOWING FORM:
                                ;NAME:   .WORD LENGTH    ;LENGTH *2
                                ;        .WORD FC0ADR    ;ADDRESS OF FUNCTION CODE 0 ROUTINE
                                ;        .WORD FC1ADR    ;ADDRESS OF FCN CODE 1 ROUTINE
                                ;        .....
                                ;        .WORD FCNADR    ;ADDRESS OF LAST FCN CODE ROUTINE.
                                ;LENGTH=.-NAME-2
                                ;
                                ;
        000000                  CAFTBL=0
        000000                  CMFTBL=0
        000000                  CCFTBL=0
                                ;
                                ;DP11 ISR
                                .IF NDF DPPRES
                                DPPRES= 0
                                .ENDC
                                .IF Z DPPRES-1
                                .GLOBL CP.OSU,CP.MDM,CP.MOD,CP.INI,CP.KLL,CP.RST
000012  000014                  CPFTBL: .WORD CPFLEN     ;LENGTH OF DP11 FUNCTION TABLE
000014  000000G                         .WORD CP.OSU     ;0=DP11 TRANSMIT ENABLE
000016  000000G                         .WORD CP.MDM     ;1=DP11 MODEM CONTROL
000020  000000G                         .WORD CP.MOD     ;2=DP11 MODE CONTROL
000022  000000G                         .WORD CP.INI     ;3=DP11 LINE INITIALIZATION
000024  000000G                         .WORD CP.RST     ;4=SYSTEM RESTART
000026  000000G                         .WORD CP.KLL     ;5=KILL I/O
        000014                  CPFLEN=.-CPFTBL-2
                                .ENDC
                                .IF Z DPPRES
                                CPFTBL= 0
                                .ENDC
```

```
                                        ;TERMINAL (TAP) TABLES
                                        ;MATRIX SWITCH ON FUNCTION CODE
                                        ;CONTAINS TABLES WITH THE SAME FUNCTION AS THOSE FOR THE ISR.
                                        ;
000030  000004                          C.SAT:   .WORD C.SATL     ;NUMBER OF TAP TYPE CODES INCLUDED
000032  000000                                   .WORD CTFTBL     ;0= ITTY TAP
                000001                  X=TAPMAX
                                        .IF NZ X
000034  000046'                                  .WORD CBFTBL     ;1= BISYNC (2780) TAP
                000000                  X=X-1
                                        .ENDC
                000004                  C.SATL= .-C.SAT-2
                                        ;
                                        ;SIB WORK ENTRY FUNCTION CODE TABLES
                                        ;C.SFJT: CONTAINS THE ADDRESS OF THE SIB ENTRY
                                        ;TABLES FOR ALL TAPS USED INT THE COMPLETE COMTEX SSYSTEM.
                                        ;IT IS USED IN THE SIB UNLOADER TO VECTOR TO THE PROPER
                                        ;SERVICE ROUTINE FOR ALL ENTRIES.
                                        ;
000036  000000                          C.SFJT:  .WORD CTSIBT     ;0= ITTY TAP
                000001                  X=TAPMAX
                                        .IF NZ X
000040  000062'                                  .WORD CBSIBT     ;1= BISYNC (2780) TAP
                000000                  X=X-1
                                        .ENDC
                                        ;
                                        ;EACH TAP SHOULD THEN HAVE A TABLE OF THE FORM:
                                        ;NAME:   .WORD LENGTH     ;LENGTH OF TABLE *2
                                        ;        .WORD TC0ADR     ;ADDRESS OF TYPE CODE 0 SERVICE ROUTINE
                                        ;        .WORD TC1ADR     ;ADDRESS OF TC1 SERVICE ROUTINE
                                        ;        .....
                                        ;        .WORD TCNADR     ;ADDRESS OF LAST SERVICE ROUTINE
                                        ;LENGTH=.-NAME-2
                                        ;
                                        ;TAP LINE TABLE LENGTH TABLE
000042  000000                          C.SLTL:  .WORD C.TLTL     ;0=ITTY LINE TABLE LENGTH
                000001                  X=TAPMAX
                                        .IF NZ X
000044  000000G                                  .WORD C.BLTL     ;1=BISYNCH TAP LINE TABLE LENGTH
                000000                  X=X-1
                                        .ENDC
                                        ;
                000000                  CTFTBL=0
                000000                  CTSIBT=0
                000000                  C.TLTL=0
                                        ;BISYNCH TAP
                                        .IF NDF BIPRES
                                        BIPRES= 0
                                        .ENDC
                                        .IF Z BIPRES-1
                                        .GLOBL C.BPD,C.BPTC,C.BARB,C.BINT,C.BLTL
                                        .GLOBL C.BMSE,C.BODE
000046  000012                          CBFTBL:  .WORD CBFLEN     ;LENGTH OF TABLE FUNCTION CODES
000050  000000G                                  .WORD C.BPD      ;0=PUT DATA
000052  000000                                   .WORD 0          ;1=UNASSIGNED
000054  000000G                                  .WORD C.BPTC     ;2=MODE CONTROL
000056  000000G                                  .WORD C.BARB     ;3=ASSIGN RECEIVE BUFFER
```

```
 402 000060  000000G                    .WORD C.BINT    ;4=LINIT
 403         000012           CBFLEN=.-CBFTBL-2
 404                          ;
 405 000062  000006           CBSIBT: .WORD CBSIBL    ;LENGTH OF BITAP SIB FUNCTION TABLE
 406 000064  000000                   .WORD 0         ;0=UNASSIGNED
 407 000066  000000G                  .WORD C.BMSE    ;1=MODEM STATUS
 408 000070  000000G                  .WORD C.BODE    ;2=TRANSMIT DONE
 409         000006           CBSIBL=.-CBSIBT-2
 410                          .ENDC
 411                          .IF Z BIPRES
 412                          CBFTBL= 0
 413                          CBSIBT= 0
 414                          C.BLTL= 0
 415                          .ENDC
 416                          ;
```

```
418                                     ;
419                                     ;
```

```
                                ;
                                        .IF Z SWB-2
                                        .GLOBL  C.SLOK,CSPSND
                                        .IFF
                                ;
                                ;
                                ;PRIORITY SCHEDULER ROUTINE CODING
                                ;ENTRANCE IS TO LOCATION NAMED C.SPS
                                
000072                          C.SPS:
000072                          CSPSND: .IF LT SWB-2
                                        RTI
                                .ENDC
                                .IF Z SWBC
000072  000207                          RTS PC                  ;CALLED FROM USER TASK ON SIB EVENT FLAG
                                ;
                                .ENDC
                                        ENDC
```

```
                                        ;THESE ARE GENERALIZED ROUTINES TO SAVE AND RESTORE THE REGISTERS.
                                        ;REGISTER SAVE ROUTINE
                                        ;ENTERED VIA JSR R5,C.SREG
000074                                  C.SREG: .IF DF RSTS
                                                .GLOBL  REGSAV
                                                JMP     REGSAV
                                                .IFF
000074  010446                                  MOV     R4,-(SP)
000076  010346                                  MOV     R3,-(SP)
000100  010246                                  MOV     R2,-(SP)
000102  010146                                  MOV     R1,-(SP)
000104  010046                                  MOV     R0,-(SP)
000106  010546                                  MOV     R5,-(SP)        ;PUT RETURN ADD ON STACK
000110  016605  000014                          MOV     14(SP),R5       ;RESTORE OLD CONTENTS OF R5
000114  000207                                  RTS     PC              ;RETURN
                                                .ENDC

                                        ;REGISTER UNSAVE ROUTINE
                                        ;ENTERED VIA JSR R5,C.SRES
000116                                  C.SRES: .IF DF RSTS
                                                .GLOBL  REGRES
                                                JMP     REGRES
                                                .IFF
000116  005726                                  TST     (SP)+
000120  012600                                  MOV     (SP)+,R0
000122  012601                                  MOV     (SP)+,R1
000124  012602                                  MOV     (SP)+,R2
000126  012603                                  MOV     (SP)+,R3
000130  012604                                  MOV     (SP)+,R4
000132  000205                                  RTS     R5
                                                .ENDC

                                .IF NZ SWB-4
                                .EOT
                                ;END OF TAPE 1 OF 5
                                .ENDC
```

```
                                .IF NDF FAKELU
                               FAKELU= 0
                                .ENDC
                                .IF NZ FAKELU
                               ;FAKE BUFFER LOADER/UNLOADER ROUTINE.
                               ;THIS ROUTINE ALLOWS DIRECT SERVICE OF THE ISR BY THE TAP
                               ;ROUTINE WHICH IS TO SERVICE THE BUFFER UNLOADER.
                               ;IT ASSUMES USE OF THE SAME TABLE AS THE SYSTEM
                               ;BUFFER UNLOADER/UNLOADER WHICH RESIDES IN THE SCIP.
                               ;A SPECIFIC USE FOR THIS LOADER/UNLOADER WOULD BE THE
                               ,SERIVCE OF A DM11 OR SIMILAR DEVICE WHERE RECEIVED CHARACTERS
                               ,HAVE BEEN BUFFERED BY THE HARDWARE.  THIS ROUTINE
                               ;COULD BE USED IF THE DEVICE WERE BEING
                               ;UNLOADED BY A TIMED ENTRY ROUTINE.
                               ;
                               ;WITH THE CURRENT TELETYPE TAP, THIS ROUTINE MUST RUN AT THE
                               ;SAME LEVEL AS THE SCIP BUFFER UNLOADER, IN ORDER
                               ;TO PREVENT POSSIBLE ERRORS IN THE ROUTINE SERVICE.
                               ;
                                       .GLOBL CFKLU
                               ;
000134  004567  177734       CFKLU:  JSR     R5,C.SREG               ;SAVE REGISTERS
000140  010500                       MOV     R5,R0           ;SAVE OFF PARAM WORD
000142  010401                       MOV     R4,R1           ;SWAP SECOND ENTRY
000144  010304                       MOV     R3,P4           ;SWAP THIRD ENTRY
000146  005003                       CLR     R3              ;LINE NUMBER IS 0
000150  016702  001040               MOV     C.STAB+0,R2     ;GET LINETABLE FOR LINE 0
000154  000305                       SWAB    R5              ;GET FUNCTION CODE
000156  006305                       ASL     R5              ;MAKE INTO WORD INDEX
                                       .IF DF RSTS
                                       JSR     PC,@CTSIBT+2(R5)        ;GO TO THE CORRECT ROUTINE
                                       .IFF
000160  004775  000064'               JSR     PC,@CBSIBT+2(R5)        ;GO TO THE CORRECT ROUTINE
                                       .ENDC
000164  004567  177726               JSR     R5,C.SRES               ;RESTORE REGISTERS
000170  000207                       RTS PC  ;AND THEN EXIT
                                .ENDC
```

```
                                        ;SCIP SYSTEM INTERFACE WITH USER PROGRAM LEVEL (UIB)
                                        ;LOADER LOADS FOUR WORDS OF PARAMETERS.  THE TYPE CODE
                                        ;IS SPECIFIED IN THE LOW ORDER 6 BITS OF THE HIGH BYTE OF
                                        ;THE FIRST WORD.  THE TWO HIGH ORDER BITS OF THAT BYTE ARE
                                        ;UTILIZED TO INDICATE THE NUMBER OF VALID WORDS IN THAT ENTRY.
                                        ;THE LINE NUMBER OCCUPIES THE LOW ORDER BYTE OF THE FIRST
                                        ;WORD OF THE ENTRY.  THIS WORD MUST NEVER BE ZERO.
                                        ;THE UNLOADER ROUTINE IS ACTUALLY THE GET EMT COMMAND RTN.

000172                                  C.SFY:  .IF NZ C.USFY
000172  005767  000540'                        TST     C.SHI   ;IS OVERFLOW FLAG SET
000176  001046                                BNE     C.SGAY  ;YES
                                        .ENDC
000200  010546                                MOV     R5,-(SP)        ;SAVE ORIGINAL PARAM POINTER
000202  010146                                MOV     R1,-(SP)        ;SAVE R1
                                        .IF NZ C.USFY
                                                .IF DF LSI11
                                                MFPS    -(SP)           ;SAVE STATUS OF CALLER
                                                MTPS    #340            ;SET PRIORITY TO LEVEL 7
                                                .IFF
000204  113746  177776                          MOVB    @#C.STAT,-(SP)  ;SAVE STATUS OF CALLER
000210  112737  000340  177776                  MOVB    #340,@#C.STAT   ;SET PRIORITY TO LEVEL 7
                                                .ENDC
000216  016701  000546'                         MOV     C.SBPY,R1       ;R1 IS BUFFER POINTER
000222  026727  000542' 000020                  CMP     C.SBCY,#C.USFY  ;ANY ENTRIES LEFT?
000230  103034                                  BHIS    C.SGBY          ;NO
000232  012521                                  MOV     (R5)+,(R1)+
000234  012521                                  MOV (R5)+,(R1)+ ;AND MOVE FOUR WORDS
000236  012521                                  MOV (R5)+,(R1)+
000240  012521                                  MOV (R5)+,(R1)+
000242  022701  000742'                         CMP     #C.SBEY,R1      ;IS IT BUFFER END
000246  103416                                  BLO     C.SFVY  ;
000250  010167  000546'                         MOV     R1,C.SBPY       ;SAVE BUFFER POINTER
000254  005267  000542'          C.SFGY: INC     C.SBCY  ;INCR INTERFACE CNTR
000260                           C.SGEY: .IF DF LSI11
                                                MTPS    (SP)+           ;RESTORE CALLER PRIORITY
                                                .IFF
000260  1'2637  177776                          MOVB    (SP)+,@#C.STAT  ;RESTORE CALLER PRIORITY
                                                .ENDC
                                        .IF Z SWB-4
000264  016701  000714                          MOV     TASKAD,R1       ;GET CURRENT TCB
000270  052761  000001  000022                  BIS     #UIBEF,T.EFLG(R1) ;SET UIB LOADED EVENT FLAG
                                        .ENDC
                                        .IF Z SWB-3
                                                MOV .CRTSK,R1   ;GET CURRENT ATL NODE
                                                BIS #UIBEF,A.EF(R1)     ;SET UIB LOADED EVENT FLAG
                                                BISB #EV.SE,.SERFG      ;AND DECLARE SIGNIFICANT EVENT
                                        .ENDC
                                        .IF Z SWB-5                     ;RT-11
                                                MOV     R0,-(SP)        ;SAVE R0
                                                JSR     PC,RESUME       ;RESUME SUSPENDED TASK
                                                MOV     (SP)+,R0        ;RESTORE R0
                                        .ENDC
                                        .ENDC
                                         IF Z C.USFY
                                                MOV R0,-(SP)            ;ALSO SAVE R0
                                                MOV @R5,R1      ;THEN GET FIRST WORD OF PLIST
```

```
                                        SWAB R1          ;AND GET HI BYTE
                                        BIC #177740,R1   ;AND DROP ALL BUT TYPE CODE
                                        ASL R1           ;MULTIPLY BY 2
                                        MOV #C.SFUJ,R0   ;THEN GET USER TABLE ADDRESS
                                        BEQ UIBXIT       ;USER FORGOT
                                        CMP R1,(R0)+     ;IS TYPE CODE SERVICED
                                        BGE UIBXIT       ;NO
                                        ADD R1,R0        ;THEN GET TYPE CODE ADDRESSED
                                        MOV @R0,R0               ;IS THAT TYPE CODE SERVICED
                                        BEQ UIBXIT       ;NO
                                        JSR PC,@R0       ;CALL USER ROUTINE
                                UIBXIT: MOV (SP)+,R0     ;RESTORE R0
                                .GLOBL C.SFUJ
                                .ENDC
000276  012601                         MOV     (SP)+,R1         ;RESTORE R1
000300  012605                         MOV     (SP)+,R5         ;RESTORE R5
000302  000207                         RTS     PC       ;RETURN TO CALLER
                                .IF NZ C.USFY
000304  012767  000552'  000546'  C.SFVY: MOV     #C.SBSY,C.SBPY  ;RESET BUFFER POINTER TO START
000312  000760                         BR      C.SFGY   ;NOW GO EXIT
000314  005267  000544'          C.SGAY: INC     C.SLEY   ;INCR INTERNAL ERROR CNTR
000320  000207                         RTS     PC       ;RETURN TO CALLER
000322  052767  000001  000540' C.SGBY: BIS     #1,C.SHI         ;SET OVRFLOW 1 FLAG IND
000330  005267  000544'                 INC C.SLEY       ;INCREMENT COUNT OF LOST ENTRIES
000334  000751                         BR C.SGEY
                                ;
                                ;
                                .IF DF RSX11
000000                          .PSECT XXX
                                .ENDC
                                ;
                                ;
```

```
                                        ;SCIP EMT GET COMMAND ROUTINE - EMT 101
                                        ;UNLOADS USER CIRCULAR BUFFER.
                                        ;ENTERED FROM MONITOR EMT TRAP HANDLER.
                                        ;PARAMETER IN R1 CONTAINS ADDRESS OF USER LOCATION
                                        ;INTO WHICH TO LOAD BUFFER ENTRIES.  IF BIT 15 OF
                                        ;THE FIRST PARAMETER WORD IS ZERO, THE FIRST ENTRY IS UNLOADED.
                                        ;IF IT IS NON-ZERO, THE LOW ORDER BYTE OF THAT
                                        ;WORD IS TAKEN AS THE LINE NUMBER, FOR WHICH
                                        ;THE BUFFER IS SEARCHED FOR AN ENTRY.   IF
                                        ;ENTRY IS FOUND, IT IS LOADED INTO THE USER
                                        ;SPECIFIED ADDRESS, AND THE BUFFER ENTRY
                                        ;WORD ONE IS SET TO ZERO.  HOWEVER, THE ENTRY IS
                                        ;NOT REMOVED, NOR IS THE COUNTER DECREMENTED.
                                        ;(THIS WOULD BE DONE BY NORMAL GET COMMAND,
                                        ;WHICH WILL DISCARD ALL ZERO TYPE CODE
                                        ;ENTRIES FOUND, AND GET FIRST GOOD ENTRY
                                        ;ENCOUNTERED)
                                        ;ON RETURN TO USER, IF ENTRY NOT AVAILABLE,
                                        ;RETURN WITH N CONDITION CODE SET.   IF ENTRY
                                        ;LOADED INTO USER ADDRESSES, RETURN TO USER
                                        ;WITH ALL CONDITION CODES RESET.   IN ALL RETURNS
                                        ;TO USER, REGS ARE RESTORED AND STACK POINTER
                                        ;IS POINTING TO ENTRY ON STACK PREVIOUS TO USER
                                        ;PUSHING EMT PARAMETER ON THE STACK.

                                                .GLOBL  C.SGET

000000  005767  000536          C.SGET: TST     C.SBCY  ;ANY ENTRIES IN BUFFER
000004  001453                          BEQ C.SFCY      ;NO ENTRIES, TELL CALLER
000006  016702  000536                  MOV     C.SFEY,R2       ;PUT BUFFER POINTER IN R2
000012  032767  000001  000520          BIT     #1,C.SHI        ;IS OVERFLOW 1 IND SET
000020  001026                          BNE     C.SGMY  ;YES
000022  042767  000003  000510          BIC     #3,C.SHI        ;CLEAR ALL OVERFLOW FLAGS
000030  005367  000506                  DEC     C.SBCY  ;DECREMENT ENTRY COUNT
                                        .IF DF LSI11
                                        MFPS    -(SP)           ;SAVE PROCESSOR PRIORITY
                                        MTPS    #340            ;;;LOCK OUT INTERRUPTS
                                        .IFF
000034  113746  177776                  MOVB    @#C.STAT,-(SP)  ;SAVE PROCESSOR PRIORITY
000040  112737  000340  177776          MOVB    #340,@#C.STAT   ;;;LOCK OUT INTERRUPTS
                                        .ENDC
000046  012221                          MOV     (R2)+,(R1)+     ;;;MOVE ENTRY TO USER STORE
000050  012221                          MOV     (R2)+,(R1)+     ;;;
000052  012221                          MOV     (R2)+,(R1)+     ;;;STORE IN USERS AREA.  4 WORDS
000054  012221                          MOV     (R2)+,(R1)+     ;;;
000056  004767  000036                  JSR     PC,C.SFQY       ;;;TEST FOR BUFFER END
000062  010267  000462                  MOV     R2,C.SFEY       ;;;SAVE BUFFER POINTER
                                        .IF DF LSI11
                                        MTPS    (SP)+           ;;;RESTORE PROCESSOR PRIORITY
                                        .IFF
000066  112637  177776                  MOVB    (SP)+,@#C.STAT  ;;;RESTORE PROCESSOR PRIORITY
                                        .ENDC
000072  000257                  C.SBGY: CLALL   ;CLEAR ALL CONDITION CODES
000074  000207                          RTS PC  ;AND RETURN TO EMT HANDLER
                                        ;
                                        ;THE UIB HAS OVERFLOWED.  TELL THE USER.
000076  052767  000002  000434  C.SGMY: BIS     #2,C.SHI        ;SET OFLO FLAG 2
```

```
                                                ;CLEAR OFLO FLAG 1
                                                ;GIVE USER OVERFLOW ERROR
                                                ;
                                ;SUBROUTINE TO TEST AND ADJUST FOR BUFFER END
                                C.SFQY:                         ;IS IT BUFFER END
                                                ;NO-GO AROUND RESET
                                                ;RESET BUFFER POINTER TO START
                                C.SFLY:                         ;RETURN TO CALLING ROUTINE
                                ;
                                ;THERE ARE NO ENTRIES IN THE UIB
                                C.SFCY:                         ;CLEAR ALL CONDITION CODES
                                                ;THEN SET N CC
                                                ;AND RETURN TO EMT HANDLER
                                ;
                                .ENDC                           ;C.USFY
                                ;EMT 100 SERVICE ROUTINE.
                                ;ENTERED FROM MONITOR TYPE EMT HANDLER
                                ;ASSUMES ENTRY VIA JSR PC WITH R1 CONTAINING USER
                                ;PARAMETER POINTER AND PS CONTAING USER PRIORITY LEVEL.

                                .GLOBL  C.SPUT
                                C.SPUT:                         ;GET FIRST WORD OF PARAMETER LIST
                                                                ;LINE # IS ALWAYS 0
                                                ;GET LINE TABLE ADDRESS
                                                ;PLACE OPCODE IN LO BYTE
                                                ;THIS IS ISR/DEVICE REQUEST
                                ;
                                ;THIS EMT 100 REQUEST IS FOR A TAP/TERMINAL
                                                ;CLEAR ALL BUT 5 RIGHTMOST BITS
                                                ;GO TO MATRIX SWITCH AND RETURN
                                                ;TO EMT HANDLER
                                ;
                                ;THIS IS A REQUEST FOR THE DEVICE (ISR) MATRIX SWITCH
                                C.SP"A:                         ;DROP TO TYPE CODE
                                                ;CALL DEVICE TYPE MATRIX SWITCH AND
                                                ;RETURN TO EMT HANDLER
                                ;
```

```
                                        .IF DF RSX11
000336                                  .PSECT ISR
                                        .ENDC
                                ;GENERAL SYSTEM MATRIX SWITCH ON FUNCTION CODE - DEVICE TYPE
                                ;USER ENTERS VIA JSR PC,C.SARP
                                ;REGISTERS SET AS FOLLOWS:
                                ;       R0=FUNCTION CODE IN 5 RIGHTMOST BITS
                                ;       R1=IF FROM EMT, CONTAINS POINTER TO USER PARAMETERS
                                ;       R2=POINTER TO LINE TABLES FOR LINE
                                ;       R3=LINE NUMBER AS A WORD INDEX
                                ;UPON RETURN TO USER, THE REGISTERS ARE RESTORED.
                                ;THE CONTENTS OF THE CONDITION CODE ARE PASSED BACK
                                ;TO USER AS RECEIVED FROM SUBROUTINES.
                                ;ALL SUBROUTINES SWITCHED TO RETURN BACK TO C.SARR

000336  011646                  C.SARP: MOV     @SP,-(SP)       ;SAVE SPACE FOR LATER
000340  004567  177530                  JSR     R5,C.SREG       ;SAVE USER REGISTERS
000344  012705  000014'                MOV     #CPFTBL+2,R5    ;HOLDS TABLE ADDRESS
000350                          C.SART:
000350  042700  177740                  BIC #177740,R0  ;GET FUNCTION CODE
000354  006300                          ASL R0  ;*2
000356  060005                          ADD R0,R5       ;THEN GET ADDRESS OF ADDRESS OF RTN
000360  011505                          MOV @R5,R5      ;THEN RTN ADDRESS
000362  004715                          JSR PC,@R5              ;AND TRANSFER TO RTN
000364  000410                          BR C.SARJ               ;AND LEAVE THIS PLACE

                                ;GENERAL SYSTEM MATRIX SWITCH ON FUNCTION CODE - TERMINAL TYPE
                                ;USER ENTERS VIA JSR PC, C.SATP
                                ;ALL OTHER CONDITIONS SAME AS DEVICE MATRIX

000366  011646                  C.SATP: MOV     @SP,-(SP)       ;SAVE SPACE FOR LATER
000370  004567  177500                  JSR     R5,C.SREG       ;SAVE USER REGISTERS
000374  012705  000050'                MOV     #CBFTBL+2,R5    ;HOLDS FIRST SWITCH START ADDR
000400  000763                          BR      C.SART  ;GO THRU COMMON CODING
                                ;
000402  000243                  C.SATN: CLVC
000404  000274                          SETNZ   ;SET ERROR RETURN
000406                          C.SARJ: .IF DF LSI11
                                        MFPS    16(SP)          ;SAVE CONDITION CODE IN SPACE
                                        .IFF
000406  013766  177776  000016          MOV     @#C.STAT,16(SP) ;SAVE CONDITION CODE IN SPACE
                                        .ENDC
000414  004567  177476                  JSR     R5,C.SRES       ;RESTORE REGS
000420  000002                          RTI     ;LOAD PC AND PS FOR USER RETURN
                                        ;NOT A RETURN FROM INTERRUPT
                                ;
                                        .IF DF RSX11
000176                                  .PSECT  XXX
                                        .ENDC
                                .IF NZ SWB-4
                                        .EOT
                                        ;END OF TAPE 3 OF 5
                                .ENDC
```

```
                                        ;EMT 103 SERVICE ROUTINE
                                        ;SYSTEM TABLE INITIALIZATION - EMT
                                        ;ENTERED FROM MONITOR TYPE EMT HANDLER
                                        ;ASSUMES ENTRY VIA JSR PC WITH R1 CONTAINING PARAMETER LIST
                                        ;ADDRESS AND ALL USER REGISTERS SAVED WITH PS AT USER LEVEL

                                                .GLOBL  C.STIN
000176                                  C.STIN: .IF LT SWB-2
                                                MOVB    #340,@#C.STAT   ;SET TO INHIBIT INTERRUPTS
                                        .ENDC
000176  011103                                  MOV     @R1,R3  ;GET PARAMETER
000200  001404                                  BEQ     C.STIA  ;STANDARD INIT
000202  004767  000544                          JSR     PC,CITB ;RESTART INIT
000206  000257                                  CCALL
000210  000207                                  RTS     PC
                                        ;
                                        ;THIS IS A STANDARD LINIT CALL
000212  005767  001212'                 C.STIA: TST     C.STRT  ;IS IT FIRST ENTRANCE
000216  001002                                  BNE     C.STIB  ;NO
000220  004767  000526                          JSR     PC,CITB ;GO INITIALIZE SCIP
000224  005003                          C.STIB:         CLR     R3      ;CLEAR R3 TO LINE NUMBER
000226  016702  001210'                          MOV C.STAK,R2   ;GET NEXT POSSIBLE  LINE TABLE ADDRESS
000232  001001                                  BNE C.STIC              ;BUFFER AVAILABLE, CONTINUE
                                                .IF Z SWB-2
                                                MOV     R1,-(SP)        ;SAVE R1 - CAUSE HAVE TO CLOBBER IT
                                                MOV     #776,R1 ;MOVE IN SIZE OFF BUFFER WANTED
                                                ADD     #BUFHDR,R1      ;ADD IN SIZE OF BUFFER HEADER
                                                .GLOBL  MONPOL
                                                MOV     #40.,R2         ;MAKE SURE SYSTEM STILL HAS SOMETHING LEFT
                                                CALLX   BUFFER,R5,MONPOL        ;GET BUFFER FROM MONITOR POOL
                                                BCS     C.STIL                  ;NO BUFFER - FATAL ERROR
                                                MOV     (SP)+,R1                ;NOW RESTORE R1
                                                ADD #BUFHDR,R4          ;STEP OVER BUFFER HEADER
                                                MOV R4,R2               ;PLACE BEGINNING BUFFER ADDRESS IN R2
                                                MOV R4,C.STAK           ;AND TEMPORARY LINE TABLE ADDRESS
                                                MOV R4,C.STAX           ;AND SAVE BEGINNING ADDRESS
                                                ADD #776,R4             ;THEN CALCULATE ENDING ADDRESS
                                                MOV R4,C.STAE           ;AND SAVE FOR LINE TABLE CHECKS
                                        C.STIK: CLR -(R4)       ;CLEAR THAT BUFFER AS OBTAINED
                                                CMP R4,R2       ;AND TEST FOR COMPLETION
                                                BHI C.STIK      ;CONTINUE UNTIL BUFFER ZEROED
                                                .ENDC
                                                .IF NZ SWB-2
000234  000440                                  BR C.STIF               ;THATS AN ERROR IN ALL SYSTEMS
                                                 ENDC
000236  010263  001214'                 C.STIC: MOV R2,C.STAB(R3)       ;AND PLACE IT IN FOR THIS LINE
000242  016104  000002                          MOV 2(R1),R4    ;AND GET TYPE CODES
000246  010412                                  MOV R4,@R2      ;AND STORE IN TAP LINE TABLE
000250  105004                                  CLRB R4
000252  000304                                  SWAB R4
000254  006304                                  ASL R4  ;THEN GET TAP CODE MULT BY 2
000256  016404  000042'                        MOV C.SLTL(R4),R4       ;GET LINE TABLE LENGTH
000262  060204                                  ADD R2,R4       ;ALSO INCREMENT TO NEXT
000264  010467  001210'                        MOV R4,C.STAK   ;OK, UPDATE NEXT AVAILABLE
000270  012700  000004                        MOV #4,R0       ;OPCODE FOR TAP INITIALIZATION
000274  004767  000366'                        JSR     PC,C.SATP       ;GO TO TAP
000300  100416                                BMI     C.STII  ;GOT AN ERROR
```

```
813 000302  005061  000004                  CLR     4(R1)   ;LOOKS GOOD
814 000306  012700  000003                  MOV     #3,R0   ;SET CODE TO GO THRU DEVICE MATRIX
815 000312  004767  000336'                 JSR     PC,C.SARP       ;GO FOR ISR INIT
816 000316  100407                          BMI C.STII      ;ERROR
817 000320  012767  177777  001212'         MOV #-1,C STRT  ;FLAG INITIALIZATION COMPLETE
818 000326  000257                          CLALL           ;CLEAR ERROR CONDITIONS
819 000330  000207                  C.STIH: RTS PC  ;RETURN TO EMT HANDLER
820 000332                          C.STID:
821 000332  000403                          BR      C.STIG
822 000334  012601                  C.STIL: MOV     (SP)+,R1        ;RESTORE R1 IN ERROR COND IN RSTS&RSX11
823 000336                          C.STIF:
824 000336  005063  001214'         C.STII: CLR C.STAB(R3)  ;CLEAR INITIALIZTION (LINE TABLE) ADDRESS
825 000342  000257                  C.STIG: CLALL   ;DROP EXTRANEOUS CONDITION CODES
826 000344  000270                          SEN     ;SET CONDITION CODE N
827 000346  000207                          RTS PC  ;RETURN TO EMT HANDLER
```

```
                                        .IF NZ CUTB
                                .GLOBL CTEMT
                                ;EMT 102 SERVICE ROUTINE
                                ;PUTTM AND DELTM
                                ;ASSUMES ENTRY FROM EMT HANDLER VIA JSR PC WITH
                                ;USER PARAMETER ADDRESS IN R1 AND PRIORITY STATUS
                                ;REGISTER SET TO USER LEVEL, ALSO ALL REGISTERS ARE
                                ;VOLATILE.
                                ;
000350  012100                  CTEMT:  MOV (R1)+,R0    ;GET FIRST PARAMETER WORD
000352  052700  177400                  BIS #177400,R0  ;FLAG THIS AS A EMT CALL IN QUEUE
000356  012104                          MOV (R1)+,R4    ;GET USER TICK COUNTER
000360  011105                          MOV @R1,R5      ;AND THEN PARAMETER WORD
                                .IF NDF TIMEDQ
                                TIMEDQ= 1       ;TIMER DEQUEUE DEFAULTS TO PRESENT
                                .ENDC
                                .IF NZ TIMEDQ
000362  032761  177400  177774          BIT #177400,-4(R1)      ;TEST FOR QUEUE/DEQUEUE
000370  001402                          BEQ CTQUE       ;ZERO IS QUEUE ROUTINE
                                ;
                                ;THIS SERVICE IS FOR A TIMER DEQUEUE
000372  000167  000526'                  JMP     CTDEL           ;CALL SUBROUTINE AND RETURN TO EMT
                                ;
                                ;THIS SERVICE IS FOR A TIMER QUEUE
000376                          CTQUE:  .ENDC
000376  000167  000422'                  JMP     CTIME   ;CALL TIMER LINK SUBROUTINE AND RETURN
                                        .ENDC
```

```
                                ;SYSTEM TIMER TAP LEVEL
                                ;ROUTINE FOR TIMING PROGRAM PRIORITY LEVEL ACTIVATION REQUESTS
                                ;ENTERED FROM PRIORITY SCHEDULER
                                        .GLOBL  CTPLR

000402                          CTPLR:
```

```
                                ;SYSTEM TIMER-LINE TIMER ROUTINE
                                ;ENTERED FROM CTPLR
                                ;EXITS BACK TO PRIORITY SCHEDULER

                                .IF Z SWB-5                     ;RT-11
                                .GLOBL RESUME,DONE
                                .ENDC

000402                          CTOCK:  .IF NZ CUTB
                                        .IF NZ SWBC
                                        TST CTIKP       ;HAS CLOCK TICKED
                                        BEQ CTOXH       ;NO EXIT
                                CTOXJ:  INC CTIC        ;INCREMENT MASTER TIMER
                                        TST CTBS        ;ANY TIMER ENTRIES
                                        BEQ CTOXJA      ;NO- GO EXIT
                                CTOS:   CMP @CTBS,CTIC  ;HAS FIRST ENTRY TIMER EXPIRED
                                        BNE CTOX        ;NO, GO EXIT
                                        .ENDC
000402  004567  000074'                 JSR R5,C.SREG   ;SAVE REGISTERS
                                .IF NZ SWBC
                                CTNE:   MOV CTBS,R0     ;GET ADDRESS OF EXPIRED ENTRY
                                        MOV 2(R0),R3    ;AND LINE NUMBER
                                        MOV 6(R0),R4    ;THEN PARAMETER CALLER SPECIFIED
                                        MOV 4(R0),CTBS  ;AND NEXT LINK POINTER
                                .ENDC
                                .IF Z SWBC
000406  012003                          MOV (R0)+,R3    ;RECOVER LINE NUMBER
000410  011004                          MOV @R0,R4      ;AND USER SPECIFIED PARAMETER WORD
                                ;**     .IF NZ SWB-5    ;NOT RT-11
000412  051040                          BIS @R0,-(R0)   ;AND TEST FOR EXPIRATION ATTEMPT
000414  001436                          BEQ CTRETZ      ;RETURN TO VACANCY CHAIN
000416  005020                          CLR (R0)+
000420  005010                          CLR @R0         ;THEN CLEAR 2,3 WORDS
000422  162700  000006                  SUB #6,R0       ;AND OBTAINS FIRST WORD
                                ;**     .ENDC
                                ;**     .IF Z SWB-5     ;RT-11
                                ;**     CLR (R0)
                                ;**     CLR -(R0)
                                ;**     SUB #4,R0
                                ;**     .ENDC
                                .ENDC
000426  004767  000650'                 JSR PC,CTPVAC   ;THEN RELINK OLD ENTRY INTO VACANCY
000432  032703  177400                  BIT #177400,R3  ;IS THIS EMT CALL
000436  001413                          BEQ CTTAP       ;NO, SYSTEM CALL
                                ;
                                ;THIS QUEUE ENTRY IS FOR A USER EMT CALL
000440  005003                          CLR     R3      ;CLEAR R3 TO LINE NUMBER
000442  052703  041000                  BIS #41000,R3   ;PLACE TYPE CODE=2, 2 WORDS
000446  010446                          MOV R4,-(SP)    ;PLACE PARAMETER ON STACK
000450  010346                          MOV R3,-(SP)    ;AND FIRST WORD
000452  010605                          MOV SP,R5       ;THEN SET UP R5 FOR UIB LOADER
000454  004767  000172'                 JSR PC,C.SFY    ;AND PLACE ENTRY IN UIB
000460  062706  000004                  ADD #4,SP       ;RESTORE STACK POINTER
000464  000403                          BR CTRETX       ;AND TRANSFER TO COMMON
                                ;
                                ;THIS ENTRY IS FOR A SYSTEM TIMER REQUEST
000466  016302  001214'          CTTAP:  MOV C.STAB(R3),R2       ;GET LINE TABLE ADDRESS
```

```
000472  004714                          JSR PC,@R4      ;AND TRANSFER TO ADDRESS
                                        .IF NZ SWBC
                                ;
                                CTRETX: TST CTBS        ;ARE THERE ANY OTHER ENTRIES
                                        BEQ CTORR       ;NO, THATS ALL
                                        CMP @CTBS,CTIC  ;HAS THE NEXT ONE ALSO EXPIRED
                                        BEQ CTNE        ;NO, THATS ALL FOR NOW
                                CTORR:  JSR R5,C.SRES   ;RESTORE REGISTERS
                                CTOX:   DEC CTIKB       ;DECREMENT INTERRUPT COUNTER
                                        BNE CTOXJ       ;THATS NOT ALL
                                CTOXA:  .ENDC
                                        .ENDC
                                        .IF NZ SWBC
                                        MOV #NOP,C.SPRT ;CLEAR REQUEST LOCATION
                                        .IF LT SWB-2
                                        MOV #NOP,C.SPIT ;AND LEVEL ACTIVE FLAG
                                        .ENDC
                                TIMXIT:
                                        JMP C.SPS       ;AND THEN TO THE  PRIORITY SCHEDULER
                                        .ENDC
                                        .IF NZ CUTB
                                        .IF NZ SWBC
                                ;
                                ;THERE ARE NO ENTRIES IN THE TIMER QUEUE.  SET CTIKB TO ZERO
                                CTOXJA: CLR CTIKB       ;AS I SAID, CLEAR CITKB
                                        BR CTOXA
                                        .ENDC
                                        .ENDC
                                        .IF Z SWBC
000474  004567  000116'        CTRETX: JSR R5,C.SRES   ;RESTORE REGISTERS
000500  012600                          MOV (SP)+,R0    ;AND RE- RESTORE R0 (JSR R0)
                                        .IF NZ SWB-5    ;NOT RT-11
000502  012716  000563                  MOV #ASTEX,@SP  ;POP STACK AND PLACE PARAMETER ON STACK
000506  104377                          EMT 377         ;AND EXIT AST VIA EMT CALL
000510  103777                          BCS     .       ;LOOP HERE IF DIRECTIVE ERROR
                                        .ENDC
                                        .IF Z SWB-5     ;RT-11
                                        RTS PC          ;EXIT FROM COMPLETION ROUTINE
                                        .ENDC
                                ;
000512  024040                  CTRETZ: CMP     -(R0),-(R0)     ;BACK UP TO START OF BUFFER ENTRY
000514  004767  000650'                  JSR     PC,CTPVAC       ;RETURN IT TO VACANCY CHAIN
000520  000765                          BR      CTRETX          ;AND EXIT
                                        .ENDC
                                        .IF NZ CUTB
                                ;SET UP TIMER SUBROUTINE
                                ;USER ENTERS WITH LINE NUMBER IN R0
                                ;EXPIRATION TIME CNT IN R4
                                ;ADDR OF TIMEOUT RTN IN R5
                                ;ENTRANCE VIA JSR PC,CTIME
                                ;REGS RESTORED ON RETURN
                                        .IF DF RSX11
000422                                  .PSECT ISR
                                        .ENDC

                                .GLOBL CTIME
000422  004567  177446          CTIME:  JSR     R5,C.SREG       ;SAVE REGS FOR USER
```

```
 978 000426  004767  000164                    JSR     PC,CTGVAC       ;GET VACANT LINK
 979 000432  005700                             TST     R0      ;IS IT VALID-NON ZERO
 980 000434  001427                             BEQ     CTIA    ;NO-RETURN TO USER
 981                                            .IF NZ SWBC
 982                                            ADD CTIKB,R4    ;ADD CTIKB OVERRUN VALUE, IN CASE
 983                                            ADD     CTIC,R4 ;ADD TIMER VALUE
 984                                            MOV     R4,(R0) ;PUT IN 1ST ENTRY WORD
 985                                            MOV     (SP),2(R0)      ;PUT LN IN 2ND
 986                                            MOV     12(SP),6(R0)    ;PUT RTN ADR IN 4TH
 987                                            JSR     PC,CTLNK        ;GO LINK
 988                                    CTIX:   .ENDC
 989                                            .IF Z SWBC
 990 000436  016720  000534'                    MOV JUNK,(R0)+          ;SET UP FIRST WORDS TO JSR R0,CTOCK
 991 000442  016720  000536'                    MOV JUNK+2,(R0)+
 992                                            .IF NZ SWB-5            ;NOT RT-11
 993 000446  010467  000526'                    MOV R4,C.MKUN           ;SET UP TIMER IN TICKS
 994                                            .ENDC
 995 000452  011620                             MOV @SP,(R0)+           ;PICK UP USER SPECIFIED LINE NO
 996 000454  016610  000012                     MOV 12(SP),@R0          ;AND LOAD USER SPECIFIED PARAMETER WORD
 997 000460  162700  000006                     SUB #6,R0               ;THEN OBTAIN BEGINNING ADDRESS
 998                                            .IF NZ SWB-5            ;NOT RT-11
 999 000464  010067  000532'                    MOV R0,C.MKAS   ;AND STORE ADDRESS
1000 000470  012746  000522'                    MOV #C.MKTM,-(SP)       ;PUT PARAMETER LIST ON STACK
1001 000474  104377                             EMT 377                 ;AND CALL RSX-11D
1002                                            .ENDC
1003                                            .IF Z SWB-5             ;RT-11
1004                                            MOV     R0,-(SP)        ;SAVE R0 DURONG MONITOR CALL
1005                                            MOV     R4,-(SP)        ;PUT ON NUMBER OF TIMER TICKS TO WAIT
1006                                            CLR     -(SP)           ;MOST SIG WORD OF TICK COUNT IS ZERO
1007                                            MOV     SP,R1           ;PUT ADDRESS OF TIMER BLOCK INTO R1
1008                                            MOV     R0,-(SP)        ;TIMER ID - ADDR OF SCIP TIMER NODE
1009                                            MOV     R0,-(SP)        ;COMPLETION ROUTINE ADDRRESS
1010                                            MOV     R1,-(SP)        ;ADDR OF TIMER TICK BLOCK
1011                                            MOV     #11000,-(SP)    ;MRKT OP CODE
1012                                            MOV     SP,R0           ;PUT ADDRESS OF TIMER BLOCK INTO R0
1013                                            .MRKT                   ;CALL RT-11
1014                                            .ENDC
1015 00    0  103404                            BCS     CTIB            ;IF OUT OF POOL NODES TAKE THIS BRANCH
1016                                            .ENDC
1017                                            .IF Z   SWB-5           ;RT-11
1018                                            ADD     #16,SP          ;CLEAN UP STACK (TAKE TIMER BLOCK OFF)
1019                                            .ENDC
1020 000500  004567  177412                    JSR     R5,C.SRES       ;RESTORE USER REGS
1021 000504  000254                            CLZN    ,RESET CC
1022 000506  000207                            RTS     PC              ;BACK TO USER
1023 000510                             CTIB:  .IF Z   SWBC
1024                                            .IF Z   SWB-5           ;RT-11
1025                                            ADD     #14,SP          ;CLEAN UP STACK
1026                                            MOV     (SP)+,R0        ;RESTORE R0 (ADDRESS OF SCIP TIMER NODE)
1027                                            .ENDC
1028 000510  004767  000134                    JSR     PC,CTPVAC       ;RETURN TO VACANCY CHAIN IF OUT OF NODES
1029                                            .ENDC
1030 000514  004567  177376             CTIA:  JSR     R5,C.SRES       ;RESTORE REGS
1031 000520  000254                            CLZN    ;RESET CC
1032 000522  000270                            SEN     ;SET N CODE
1033 000524  000207                            RTS     PC      ;RETURN TO USER
1034                                            .IF DF RSX11
```

```
1035 000526                             .PSECT ISR
1036                                    .ENDC
```

```
1038                                    .IF NZ SWBC
1039                                ;LINK TIME ENTRY INTO TIMER CHAIN
1040                                ;ON ENTRANCE R0=ADDRESS OF ENTRY TO LINK
1041                                ;ENTERED VIA JSR PC,CTLNK
1042                                ;ROUTINE IS RE-ENTRANT
1043
1044                                CTLNK:
1045                                    .IF NZ TLOCK
1046                                .IF NZ SLOCK
1047                                    MOV C.SLOK,-(SP)        ;SAVE SOFTWARE LOCK LEVEL
1048                                    MOV #RTI,C.SLOK ;AND SET IT ACTIVE
1049                                .ENDC
1050                                .IF Z SLOCK
1051                                    MOVB    @#C.STAT,-(SP)  ;SAVE PS PRIORITY
1052                                    MOVB    #340,@#C.STAT   ;SET PRIORITY LEVEL TO 7
1053                                .ENDC
1054                                    .ENDC
1055                                    TST     CTBS    ;IS CNTR EQUAL 0
1056                                    BEQ     CTLZ    ;YES- GO LINK ONLY ENTRY
1057                                    MOV     #CTBS-4,R4      ;SET UP LINK - IN ADDR POINTER
1058                                    MOV     CTIC,R3 ;GET TIMER VALUE
1059                                    MOV     (R0),R1 ;GET ENTRY TICK CNTR
1060                                    SUB     R3,R1   ;GET ENTRY TO LINK REL CLOCK VALUE
1061                                    MOV     CTBS,R2 ;SET TO FIRST LINK ADDR
1062                                CTLA:   MOV     (R2),R5 ;GET FIRST LINK TICK CNTR
1063                                    SUB     R3,R5   ;GET LINK REL CLOCK VALUE
1064                                    CMP     R5,R1   ;IS LINK GREATER THAN ENTRY TO LINK
1065                                    BHI     CTLE    ;GO LINK - YES
1066                                    MOV     R2,R4   ;SAVE PREV LINK ADDR
1067                                    MOV     4(R2),R2        ;GET NEXT LINK ADDR
1068                                    BNE     CTLA    ;NOT CHAIN END, LOOK AT NEXT
1069                                    MOV     R0,4(R4)        ;LINK IN AT CHAIN END
1070                                CTLC:   CLR     4(R0)
1071                                CTLBA:  .IF NZ TLOCK
1072                                    .IF NZ SLOCK
1073                                    MOV (SP)+,C.SLOK        ;RESTORE SOFTWARE LOCK STATE
1074                                    MOV @SP,-(SP)   ;MAKE A FAKE INTERRUPT
1075                                    MOV @#C.STAT,2(SP)      ;INSTEAD OF SUBROUTINE RETURN
1076                                    JMP C.SPS       ;AND CALL PRIORITY SCHEDULER
1077                                .ENDC
1078                                .IF Z SLOCK
1079                                    MOVB    (SP)+,@#C.STAT  ;RESTORE USER PRIORITY
1080                                    RTS     PC      ;RETURN TO USER
1081                                .ENDC
1082                                .ENDC
1083                                .IF Z TLOCK
1084                                    RTS     PC      ;EXIT TO CALLER
1085                                .ENDC
1086                                ;
1087                                CTLE:   MOV     4(R4),4(R0)     ;GET NXT ADDR FROM PR. , PUT IN NEW
1088                                    MOV     R0,4(R4)        ;LINK NEW ENTRY TO PRLV
1089                                    BR      CTLBA
1090                                CTLZ:   MOV     R0,CTBS
1091                                    BR      CTLC
1092                                    .ENDC
```

```
                                .IF NZ TIMEDQ
                                ;THIS SUBROUTINE PERFORMS THE ACTUAL DEQUEUE FOR THE TIMER
                                ;QUEUE ENTRIES.
                                ;ENTRY CONDITIONS ARE
                                ;R0=    LINE NUMBER AS IN TIMER QUEUE ENTRY
                                ;R4=    UNUSED
                                ;R5=    PARAMETER WORD, AS IN TIMER QUEUE ENTRY
                                ;
                                .GLOBL CTDEL
000526  004567  177342          CTDEL:  JSR R5,C.SREG   ;SAVE REGISTERS
                                .IF NZ TLOCK
                                .IF NZ SLOCK
                                        MOV C.SLOK,-(SP)        ;SAVE SOFTWARE LOCK STATE
                                        MOV #RTI,C.SLOK ;AND SET LEVEL BUSY
                                .ENDC
                                .IF Z SLOCK
                                        .IF DF LSI11
                                        MFPS    -(SP)           ;SAVE PRIORITY SCHEDULER
                                        MTPS    #340            ;AND LOCKOUT INTERRUPTS
                                        .IFF
000532  113746  177776                  MOVB @#C.STAT,-(SP)     ;SAVE PRIORITY LEVEL
000536  112737  000340  177776          MOVB #340,@#C.STAT      ;AND LOCKOUT INTERRUPTS
                                        .ENDC
                                .ENDC
                                .ENDC
                                        .IF NZ SWBC
                                        MOV #CTBS-4,R3
                                        MOV CTBS,R1     ;GET BEGINNING OF CHAIN
                                CTDEL1: BEQ CTDRET      ;NONE PRESENT
                                        CMP 2(SP),2(R1) ;IS THIS THE LINE NUMBER
                                        BNE CTDEL2      ;NO
                                        CMP 14(SP),6(R1)        ;IS THIS THE PARAMETER WORD
                                        BNE CTDEL2      ;NO
                                        MOV R1,R0
                                        MOV 4(R1),R1
                                        JSR PC,CTPVAC   ;PLACE LINK IN VACANCY QUEUE
                                        MOV R1,4(R3)
                                        BR CTDEL1       ;AND CONTINUE SEARCH
                                CTDEL2: MOV R1,R3       ;SAVE LAST ENTRY LOCATION
                                        MOV 4(R1),R1
                                        BR CTDEL1       ;AND SEARCH
                                        .ENDC
                                        .IF Z SWBC
000544  012700  000704'                 MOV #CTLB,R0            ;SEARCH ENTIRE TABLE FROM BEGINNING
000550  026660  000014  000006  CTDEL1: CMP 14(SP),6(R0)        ;;;THEN COMPARE PARAMETER WORD
000556  001004                          BNE CTDEL2              ;;;NOT SPECIFIED ENTRY
                                ;**     .IF Z SWB-5             ;RT-11
                                ;**     MOV     R0,C.CUID       ;;;TIMER ID IS NODE ADDRESS
                                ;**     MOV     R0,-(SP)        ;SAVE R0
                                ;**     .CMKT   #C.CUID         ;CALL RT-11   CANCEL TIMER
                                ;**     MOV     (SP)+,R0        ,RESTORE R0
                                ;**     JSR     PC,CTPVAC       ;RETURN NODE TO POOL
                                ;**     .ENDC
                                ;**     .IF NZ SWB-5            ;NOT RT-11
000560  005060  000004                  CLR 4(R0)               ;;;FLAG ENTRY DELETED
000564  005060  000006                  CLR 6(R0)               ;;;DITTO
                                ;**     .ENDC
```

```
1151 000570  062700  000010          CTDEL2: ADD #10,R0              ;;;THEN JUMP TO NEXT
1152 000574  020027  001204'                  CMP R0,#CTEB            ;;;IS THIS END OF LINKS
1153 000600  103763                           BLO CTDEL1              ;;;NO, LOOK FOR MODE DELETIONS
1154                                          .ENDC
1155 000602                          CTDRET:  .IF NZ TLOCK
1156                                          .IF NZ SLOCK
1157                                          MOV (SP)+,C.SLOK
1158                                          JSR R5,C.SRES   ;RESTORE REGISTERS
1159                                          MOV @SP,-(SP)   ;AND START BUILDING FAKE RTI
1160                                          CLZN    ;SHOW NO ERRORS ON RETURN
1161                                          MOV @#C.STAT,2(SP) ;
1162                                          JMP C.SPS       ;AND CALL PRIORITY SCHEDULER TO ENTER RTNS IF NECC
1163                                  .ENDC
1164                                  .IF Z SLOCK
1165                                          .IF DF LSI11
1166                                          MTPS    (SP)+           ;ALLOW INTERRUPTS
1167                                          .IFF
1168 000602  112637  177776                   MOVB (SP)+,@#C.STAT     ;ALLOW INTERRUPTS
1169                                          .ENDC
1170 000606  004567  177304                   JSR R5,C.SRES   ;AND RESTORE REGISTERS
1171 000612  000254                           CLZN    ;AND FLAG AS OK
1172 000614  000207                           RTS PC
1173                                  .ENDC
1174                                  .ENDC
1175                                  .IF Z TLOCK
1176                                          RTS     PC      ;RETURN TO CALLER
1177                                  .ENDC
1178                                  .ENDC
```

```
1180                                            .IF DF RSX11
1181 000616                                     .PSECT ISR
1182                                            .ENDC
1183                                    ;GET TIMER BUFFER VACANCY ROUTINE
1184                                    ;ENTERED VIA JSR PC,CTGVAC
1185                                    ;RETURN TO USER WITH VACANT ENTRY ADDR IN R0
1186                                    ;IF NO ENTRY AVAILABLE, RETURN TO USER WITH R0 EQUAL 0
1187                                    ;ROUTINE IS RE-ENTRANT
1188
1189 000616                             CTGVAC: .IF DF LS111
1190                                            MFPS    -(SP)   ;SAVE PS PRIORITY
1191                                            MTPS    #340    ;SET PRIORITY TO 7
1192                                            .IFF
1193 000616  113746  177776                    MOVB    @#C.STAT,-(SP)  ;SAVE PS PRIORITY
1194 000622  112737  000340  177776            MOVB    #340,@#C.STAT   ;SET PRIORITY TO 7
1195                                            .ENDC
1196 000630  016700  000046                    MOV     CTSV,R0 ;NEXT BUFFER ADDR INTO R0
1197 000634  001402                            BEQ     CTGVEX  ;YES-NO ENTRY AVAILABLE
1198 000636  011067  000040                    MOV     (R0),CTSV       ;MOV NEXT LINK ADDR TO VAC START
1199 000642                             CTGVEX: .IF DF LS111
1200                                            MTPS    (SP)+           ;RESTORE PRIORITY
1201                                            .IFF
1202 000642  112637  177776                    MOVB    (SP)+,@#C.STAT  ;RESTORE PRIORITY
1203                                            .ENDC
1204 000646  000207                            RTS     PC      ;BACK TO USER
1205
1206                                    ;RELINK ENTRY TO VACANCY CHAIN
1207                                    ;ENTERED VIA JSR PC,CTPVAC, ENTRY ADDR IN R0
1208                                    ;ROUTINE IS RE-ENTRANT
1209
1210 000650                             CTPVAC: .IF DF LS111
1211                                            MFPS    -(SP)           ;SAVE PS PRIORITY
1212                                            MTPS    #340            ;SET PRIORITY TO 7
1213                                            .IFF
1214 000650  113746  177776                    MOVB    @#C.STAT,-(SP)  ;SAVE PS PRIORITY
1215 000654  112737  000340  177776            MOVB    #340,@#C.STAT   ;SET PRIORITY TO 7
1216                                            .ENDC
1217 000662  016710  000014                    MOV CTSV,@R0    ;LINK OLD FIRST TO NEW LINK
1218 000666  010067  000010                    MOV R0,CTSV     ;AND PLACE NEW AT FIRST OF CHAIN
1219                                            .IF DF LS111
1220                                            MTPS    (SP)+           ;RESET PS PRIORITY
1221                                            .IFF
1222 000672  112637  177776                    MOVB    (SP)+,@#C.STAT  ;RESET PS PRIORITY
1223                                            .ENDC
1224 000676  000207                            RTS     PC      ;RETURN TO USER
```

```
                                        ;SYSTEM TIMER VALUES, MASTER TIMER, CHAIN POINTERS, ETC.
                                                .IF Z SWB-2
                                                .GLOBL  CTIKB,CTMAXB,CTIC,CTBS,CTSV,CTLB,CTEB
                                                .GLOBL  C.STAX,C.STAK,C.STRT,C.STAB,C.STAP,C.STAE
                                                .IFF
                                        ;
                                                .IF NZ SWBC
                                        ;
                                        ;
                                        CTIKB:  .WORD 0 ;INTERRUPT COUNTER FOR QUEUE SERVICE ROUTINE
                                        CTMAXB: .WORD 0         ;**** MAXIMUM CTIKB OVERRUN VALUE
                                        CTIC:   .WORD 0 ;MASTER TIMER COUNTER
                                                .ENDC
000700  000000                          CTBS:   .WORD 0 ;START OF ACTIVE LINK CHAIN
000702  000000                          CTSV:   .WORD   0       ;VACANCY START ADDR
000704  000000                          CTLB:   .WORD   0       ;FIRST BUFFER WORD
        001204'                         .=.+CUTB+CUTB+CUTB+CUTB+CUTB+CUTB+CUTB+CUTB-2 ;ALLOCATES BUFFER
001204                                  CTEB:
                                        ;
                                                .IF Z SWBC
                                                .IF NZ SWB-5    ;NOT RT-11
001204                                          .PSECT ISR
                                                .GLOBL  TASKAD
001204  000000                          TASKAD: .WORD 0                 ;CURRENT TASK NODE
                                                .ENDC
                                                .IF Z   SWB-5
                                                .GLOBL  TASKAD
                                        TASKAD: .WORD   0               ;TASK'S IMPURE AREA
                                                .ENDC
                                                .ENDC
                                        ;SYSTEM LINE TABLE START ADDRESS POINTER TABLE
                                        ;SIZE OF TABLE DETERMINED BY USER AT ASSEMBLY TIME
                                        ;TABLE SET WITH POINTERS AT SYSTEM INITIALIZATION TIME

001206  002216'                         C.STAE: .WORD C.STAP            ;END OF AVAILABLE LINE TABLE SPACE
001210  000000                          C.STAK: .WORD 0 ;NEXT AVAILABLE LT SPACE
001212  000000                          C.STRT: .WORD 0 ;"FIRST ENTRY" FLAG FOR INIT SERVICE
001214  000000                          C.STAB: .WORD 0 ;LINE TABLE ADDRESS TABLE
        001216'                         .=.+C.USNL+C.USNL-2
                                        ;
                                        .IF NZ C.USRT
                                        ;SYSTEM LINE TABLES
                                        ;SIZE OF AREA RESERVED DETERMINED BY USER AT ASSEMBLY TIME
                                        ;TABLES INITIALIZED FOR ASSIGNED DEVICES AT SYSTEM INITIALIZATION TIME

                                                .GLOBL C.STAR
001216  002216'                         C.STAR:.=.+C.USRT+C.USRT
                                        .ENDC
002216                                  C.STAP:
                                                .ENDC
                                        ;
                                                .IF Z SWBC
                                                .IF NZ SWB-5    ;NOT RT-11
000522                                          .PSECT XXX
                                        ;
                                        ;RSX-11D DIRECTIVE BLOCKS
        000563                          ASTEX=  563                     ;AST EXIT DPB (.BYTE 115.,1)
```

```
                        000433                  CMKTM=  433                     ;CANCEL MARK TIME DPB (.BYTE 27.,1)
                                                ;
000522    027     005   C.MKTM: .BYTE 23.,5       ;MARK TIME DPB
000524  000000                  .WORD 0
000526  000000          C.MKUN: .WORD 0           ;UNITS OF TIME
000530  000001                  .WORD 1           ;UNITS = TICKS
000532  000000          C.MKAS: .WORD 0           ;TIMER AST ENTRY POINT
                                .ENDC

                                .IF Z SWB-5                     ;RT-11
                        ,CANCEL MARK TIME PARAMETER BLOCK
                        C.CMKT: .BYTE   0,23                    ;OP CODE - CANCEL MARK TIME
                        C.CUID: .WORD   0                       ;ID WORD
                                .WORD   0                       ;NO TIME REQUIRED
                                .ENDC
                                ;
000534  004037  000402' JUNK:   JSR R0,@#CTOCK          ;FIRST WORD OF TIMER LINK
                                .ENDC
                        .IF NZ C.USFY
                        ;USER INTERFACE BUFFER AND ITS RELATED ENTRY PTRS
                        ;AND FLAGS.

000540  000000          C.SHI:  .WORD 0 ;OVERFLOW INDICATOR
000542  000000          C.SBCY: .WORD 0 ;ENTRY COUNTER
000544  000000          C.SLEY: .WORD 0 ;LOST ENTRY COUNTER
                                .IF NZ UIBSRH
                        C.SBWY: .WORD 0 ;TEMPORARY FOR SEARCH MODE
                                .ENDC
000546  000552'         C.SBPY: .WORD C.SBSY    ;ENTRY INSERTION POINTER
000550  000552'         C.SFEY: .WORD C.SBSY    ;ENTRY REMOVAL POINTER
                                ;
                        ;THIS IS THE UIB
000552  000000          C.SBSY: .WORD 0
                        X=      C.USFY+C.USFY
                                000742'
                        .=.+X+X+X+X-12
000742  000000  000000  000000  C.SBEY: .WORD 0,0,0,0
000750  000000
                                .ENDC
```

```
                                        ;INITIALIZATION SUBROUTINE - CITB
                                        ;THIS SUBROUTINE INITIALIZES ALL SCIP PARAMETERS
                                        ;FOR RESTART CAPABILITY.  IT THE CALLS THROUGH THE DEVICE
                                        ;TYPE  MATRIX SWITCH ANY ISRS WHICH INCLUDE RESTART
                                        ;ROUTINES.
                                        ;R1 AND R3 MUST BE PRESERVED OVER THIS SUBROUTINE ****
                                        ;
000752  005067  001212'          CITB:   CLR C.STRT      ;FLAG RESTART IN PROGRESS
                                        .IF NZ CUTB
                                        ;
                                        ;IF TIMER QUEUE PRESENT, REINITIALIZE ALL ENTRIES
                                        ;
                                        .IF Z   SWBC
                                        .IF DF LS111
                                        MFPS    -(SP)           ;SAVE PS
                                        MTPS    #340            ;;;LOCK OUT INTERRUPTS
                                        .IFF
000756  113746  177776                  MOVB    @#C.STAT,-(SP)  ;SAVE PS
000762  112737  000340  177776          MOVB    #340,@#C.STAT   ;;;AND LOCK OUT INTERRUPTS
                                        .ENDC
000770  012705  000710'                 MOV     #CTLB+4,R5      ;;;GET ADDR OF TIMER CHAIN + 4 BYTES
000774  005025                  CITAA:  CLR     (R5)+           ;;;FLAG ENTRY DELETED
000776  005025                          CLR     (R5)+           ;;; ...
001000  062705  000004                  ADD     #4,R5           ;;;POINT TO NEXT ENTRY + 4
001004  020527  001210'                 CMP     R5,#CTEB+4      ;;;HAVE WE DELETED THEM ALL?
001010  103771                          BLO     CITAA           ;;;NO
                                        .IF DF LS111
                                        MTPS    (SP)+           ;;;YES ALLOW INTERRUPTS
                                        .IFF
001012  112637  177776                  MOVB    (SP)+,@#C.STAT  ;;;YES, ALLOW INTRRUPTS
                                        .ENDC
                                        .IF NZ SWB-5            ;NOT RT-11
001016  012746  000433                  MOV     #CMKTM,-(SP)    ;DPB FOR CANCEL MARK TIME
001022  104377                          EMT     377             ;CALL RSX-11D/M
001024  103777                          BCS     .               ;BLEW IT!!!
                                        .ENDC
                                        .IF Z SWB-5             ;RT-11
                                        MOV     #C.CMKT,R0      ;MOVE PNTR TO PARMS LIST INTO R0
                                        CLR     C.CUID          ;CANCEL ALL TIMERS
                                        .CMKT                   ;CALL RT-11
                                        .ENDC
                                        .ENDC
001026  012702  000030                  MOV #CUTB,R2    ;GET NUMBER OF ENTRIES
001032  012705  000704'                 MOV #CTLB,R5    ;AND START ADDRESS
001036  010567  000702'                 MOV R5,CTSV     ;INITIALIZE VACANCY CHAIN ADDRESS
001042  005067  000700'                 CLR CTBS        ;AND ACTIVE CHAIN POINTER
                                        .IF NZ SWBC
                                        CLR CTIC        ;THEN MASTER TICK COUNTER
                                        CLR CTIKB       ;AND INTERRUPT COUNTER
                                        .ENDC
001046  010504                  CITA:   MOV R5,R4       ;GET ENTRY ADDRESS
001050  062705  000010                  ADD #10,R5      ;THEN NEXT LINK ADDRESS
001054  010524                          MOV R5,(R4)+    ;FIRST WORD IS POINTER TO NEXT AVAIL LINK
001056  005024                          CLR (R4)+       ;THEN CLEAR ALL WORDS OF ENTRY
001060  005024                          CLR (R4)+
001062  005024                          CLR (R4)+
001064                                  SOB     R2,CITA         ;LOOP UNTIL DONE
```

```
1378 001070  005064  177770                    CLR -10(R4)     ;THATS ALL CLEAR LAST LINK POINTER
1379                                           .ENDC
1380                                    ;
1381                                    ;IF PRIORITY LEVEL SCHEDULING PRESENT, CLEAR CLOCK
1382                                           .IF NZ CUTNPS
1383                                           CLR CTIKA
1384                                           MOV #CUTNPS,R2  ;GET COUNT OF PROGRAMS TO SCHEDULE
1385                                           MOV #CTPAD,R5   ;GET BEGINNING ADDRESS
1386                                    CITC:  MOV 2(R5),@R5   ;SET COUNTERS BACK TO ORINGINAL VALUE
1387                                           ADD #10 R5      ;AND INCREMENT TO NEXT
1388                                           DEC R2  ;DECREMENT COUNTER
1389                                           BNE CITC        ;CONTINUE
1390                                           .ENDC
1391                                    ;
1392                                    ;CLEAR UIB POINTERS AND BUFFER
1393                                    .IF NZ C.USFY
1394 001074  012705  000552'                    MOV #C.SBSY,R5  ;GET BUFFER ADDRESS
1395 001100  010567  177442                    MOV R5,C.SBPY   ;AND RESET INSERTION POINTER
1396 001104  010567  177440                    MOV R5,C.SFEY   ;AND REMOVAL POINTER
1397 001110  005067  177424                    CLR C.SHI       ;THEN CLEAR OVERFLOW POINTER
1398 001114  005067  177422                    CLR C.SBCY      ;AND ENTRY COUNTER
1399 001120  005067  177420                    CLR C.SLEY      ;AND LOST ENTRY COUNTER
1400 001124  005025                     CITD:  CLR (R5)+       ;THEN CLEAR BUFFER
1401 001126  020527  000742'                    CMP R5,#C.SBEY  ;IS THAT ALL
1402 001132  101774                            BLOS CITD       ;NO, CONTINUE
1403                                    .ENDC
1404                                    ;
1405                                    ;CLEAR LINE TABLE SPACE AND LINE TABLE POINTERS
1406 001134  012705  001214'                    MOV #C.STAB,R5  ;GET LINE TABLE ADDRESS SPACE
1407                                           .IF NZ C.USRT
1408 001140  012767  001216' 001210'           MOV #C.STAR,C.STAK      ;AND SET ALL OF SPACE AVAILABLE
1409                                           .ENDC
1410                                           .IF Z C.USRT
1411                                           CLR C.STAK              ;FORCE TABLE ADDRESS ZERO
1412                                           .ENDC
1413 001146  005025                     CITE:  CLR (R5)+       ;AND CLEAR ALL OF SPACE
1414 001150  020527  002216'                    CMP R5,#C.STAP  ;IS THAT ALL
1415 001154  103774                            BLO CITE        ;NO, CONTINUE
1416                                           .IF Z SWB-2
1417                                           MOV C.STAX,R4           ;IS THERE A BIG BUFFER OUTSTANDING
1418                                           BEQ CITI                ;NO, THATS ALL
1419                                           SUB     #BUFHDR,R4      ;MOVE BUFFER BACK TO ACTUAL BEGINNING
1420                                                                   ;THE SAVED EXACT COUNT
1421                                           BUFFER RETURN           ;YES, RETURN THE BUFFER
1422                                    CITI:  CLR C.STAX              ;THEN FORCE ADDRES ZERO
1423                                           .ENDC
1424                                    ;
1425                                    ;NOW REINITIALIZE ANY ISRS WHICH NEED IT
1426 001156  016704  000000'                    MOV C.SAR,R4    ;GET NUMBER OF ISRS
1427 001162  006204                            ASR R4  ;AS REAL LIVE NUMBER
1428 001164  012702  001212'                    MOV #C.STRT,R2
1429 001170  005012                            CLR @R2 ;START WITH ISR #)
1430 001172  012700  000004                    MOV #4,R0       ;AND CALL RESTART CODE
1431 001176  004767  000336'             CITG:  JSR PC,C.SARP   ;USING DEVICE-TYPE MATRIX SWITCH
1432 001202  005212                            INC @R2 ;GO TO NEXT ISR
1433 001204  020412                            CMP R4,@R2      ;IS THAT ALL
1434 001206  001373                            BNE CITG        ;NO, CONTINUE ISR CALLS
```

```
                                        ;
                                                .IF Z SWB-4
001210  016767  000000G 001204'                MOV     $TKTCB,TASKAD   ;SAVE OUR TCB ADDRESS
                                                .ENDC
                                                .IF Z SWB-3
                                                MOV .CRTSK,TASKAD
                                                .ENDC
                                                .IF Z SWB-5
                                                .GLOBL  RMON,CNTXT
                                                MOV     @#RMON,R4       ;GET RESIDENT MONITOR TABLE
                                                MOV     CNTXT(R4),TASKAD        ;SAVE POINTER TO IMPURE AREA
                                                .ENDC
                                        ;THATS ALL OF INITIALIZATION
                                        ;
                                        ;THIS IS A FUNNY ADDRESS USED TO FAKE EXITS FOR PROGRAM
                                        ;COMPATIBILITY FOR RETURNS TO C.SARR,CTRET,AND ALL
                                        ;COMMON ENTRANCE EXIT ROUTINES
001216                                  C.SARR:
001216                                  CTRET:
001216                                  C.SECA:
001216                                  C.SECB:
001216                                  C.SECD:
001216                                  C.SECE:
001216                                  C.SECF:
001216                                  C.SECG:
001216  000207                          C.SECH: RTS PC
                                        ;
                                                .ENDC
000001                                  .END
```

SYMBOL TABLE

```
ASTEX = 000563          CTDEL   000526RG  002 C.SECF  001216RG  003 EMTVEC= 000000        R$$JS1= 000001
A$$CHK= 000000          CTDEL1  000550R   002 C.SECG  001216RG  003 E$$DVC= 000000        R$$KWC= 000000
A$$CPS= 000000          CTDEL2  000570R   002 C.SECH  001216RG  003 E$$NSI= 000000        R$$K11= 000001
A$$NSI= 000000          CTDRET  000602R   002 C.SFCY  000134R   003 E$$PER= 000000        R$$NDC= 000003
A$$PRI= 000000          CTEB    001204R   002 C.SFEY  000550R   003 E$$XPR= 000000        R$$NDH= 000226
A$$TRP= 000000          CTEMT   000350RG  003 C.SFGY  000254R   002 FAKELU= 000001        R$$NDL= 000001
BIPRES= 000001          CTFTBL= 000000        C.SFJT  000036R   002 F$$LVL= 000001        R$$SND= 000000
BIT00 = 000001          CTGVAC  000616R   002 C.SFLY  000132R   003 G$$TPP= 000000        R$$11M= 000000
BIT01 = 000002          CTGVEX  000642R   002 C.SFQY  000120R   003 G$$TTK= 000000        SCC   = 000277
BIT02 = 000004          CTIA    000514R   002 C.SFVY  000304R   002 HALT  = 000000        SEC   = 000261
BIT03 = 000010          CTIB    000510R   002 C.SFY   000172RG  002 ISRMAX= 000003        SEN   = 000270
BIT04 = 000020          CTIME   000422RG  002 C.SGAY  000314R   002 I$$RAR= 000000        SETALL= 000277
BIT05 = 000040          CTLB    000704R   002 C.SGBY  000322R   002 I$$RDN= 000000        SETNZ = 000274
BIT06 = 000100          CTOCK   000402R   003 C.SGET  000000RG  003 JUNK    000534R  003 SETNZC= 000275
BIT07 = 000200          CTPLR   000402RG  003 C.SGEY  000260R   002 K$$CNT= 172542        SEV   = 000262
BIT08 = 000400          CTPVAC  000650R   002 C.SGMY  000076R   003 K$$CSR= 172540        SEZ   = 000264
BIT09 = 001000          CTQUE   000376R   003 C.SHI   000540R   003 K$$LDC= 000024        SIBEF = 000040
BIT10 = 002000          CTRET   001216RG  003 C.SLEY  000544R   003 K$$TPS= 000764        SLOCK = 000000
BIT11 = 004000          CTRETX  000474R   003 C.SLTL  000042R   002 LD$DB  = 000000        SWB   = 000004
BIT12 = 010000          CTRETZ  000512R   003 C.SMAX= 070000        LD$DK  = 000000        SWBC  = 000000
BIT13 = 020000          CTSIBT= 000000        C.SPS   000072RG  002 LD$DS  = 000000        S$$WPC= 000036
BIT14 = 040000          CTSV    000702R   002 C.SPTA  000166R   003 LD$DT  = 000000        S$$WPR= 000005
BIT15 = 100000          CTTAP   000466R   003 C.SPUT  000142RG  003 LD$LP  = 000000        S$$WRG= 000000
CAFTBL= 000000          CUTB  = 000030        C.SREG  000074RG  002 LD$MM  = 000000        S$$YSZ= 024000
CBFLEN= 000012          CUTNPS= 000000        C.SRES  000116RG  002 LD$PP  = 000000        TAPMAX= 000001
CBFTBL  000046R     002 C$$CKP= 000000        C.STAB  001214RG  002 LD$PR  = 000000        TASKAD  001204RG    002
CBSIBL= 000006          C$$ORE= 004000        C.STAE  001206R   002 L$$ASG= 000000        TIMEDQ= 000001
CBSIBT  000062R     002 C$$RSH= 177564        C.STAK  001210R   002 L$$DRV= 000000        TLOCK = 000001
CCC   = 000257          C.BARB= ****** G      C.STAP  002216R   002 L$$P11= 000001        TS.BLK= 170700
CCFTBL= 000000          C.BINT= ****** G      C.STAR  001216RG  002 L$$11R= 000000        TS.CKP= 000200
CFKLU   000134RG    002 C.BLTL= ****** G      C.STAT= 177776        M$$CRB= 000124        TS.CKR= 000100
CITA    001046R     003 C.BMSE= ****** G      C.STIA  000212R   003 M$$CRX= 000000        TS.EXE= 100000
CITAA   000774R     003 C.BODE= ****** G      C.STIB  000224R   003 M$$EXT= 000000        TS.MSG= 020000
CITB    000752R     003 C.BPD = ****** G      C.STIC  000236R   003 M$$FCS= 000000        TS.NRP= 010000
CITD    001124R     003 C.BPTC= ****** G      C.STID  000332R   003 M$$MGE= 000000        TS.OUT= 000400
CITE    001146R     003 C.MKAS  000532R   003 C.STIF  000336R   003 M$$MUP= 000000        TS.RDN= 040000
CITG    001176R     003 C.MKTM  000522R   003 C.STIG  000342R   003 M$$OVR= 000000        TS.RUN= 004000
CKW11 = 000000          C.MKUN  000526R   003 C.STIH  000330R   003 NOP   = 000240        T$$ACR= 000000
CLALL = 000257          C.SAR   000000R   002 C.STII  000336R   003 N$$LDV= 000001        T$$BTW= 000000
CLC   = 000241          C.SARJ  000406R   002 C.STIL  000334R   003 N$$MOV= 000021        T$$BUF= 000000
CLN   = 000250          C.SARL= 000010        C.STIN  000176RG  003 OFFSET= 000000        T$$CCA= 000000
CLV   = 000242          C.SARP  000336RG  002 C.STRT  001212R   002 PLF   = 000000        T$$CCO= 000000
CLVC  = 000243          C.SARR  001216RG  003 C.TLTL= 000000        P$$D70= 000000        T$$CTR= 000000
CLZ   = 000244          C.SART  000350R   002 C.USFY= 000020        P$$GMX= 000000        T$$C11= 000001
CLZN  = 000254          C.SAT   000030R   002 C.USNL= 000001        P$$LAS= 000000        T$$GMC= 000000
CMFTBL= 000000          C.SATL= 000004        C.USP = 000300        P$$P11= 000001        T$$GTS= 000000
CMKTM = 000433          C.SATN  000402R   002 C.USRT= 000400        P$$P45= 000000        T$$J16= 000001
CPFLEN= 000014          C.SATP  000366RG  002 DPPRES= 000001        P$$RFL= 000000        T$$KMG= 000000
CPFTBL  000012R     002 C.SBCY  000542R   003 D$$BUG= 177514        P$$RTY= 000000        T$$LWC= 000000
CP.INI= ****** G        C.SBEY  000742R   003 D$$H11= 000002        P$$R11= 000001        T$$MAN= 006307
CP.KLL= ****** G        C.SBGY  000072R   003 D$$IAG= 000000        P$$SRF= 000000        T$$RNE= 000000
CP.MDM= ****** G        C.SBPY  000546R   003 D$$ISK= 000000        Q$$OPT= 000017        T$$RPR= 000000
CP.MOD= ****** G        C.SBSY  000552R   003 D$$L11= 000001        RSX11 = 000001        T$$RST= 000000
CP.OSU= ****** G        C.SECA  001216RG  003 D$$M11= 000001        R$$DER= 000000        T$$RUB= 000000
CP.RST= ****** G        C.SECB  001216RG  003 D$$SHF= 000000        R$$EXV= 000000        T$$SYN= 000000
CSPSND  000072R     002 C.SECD  001216RG  003 D$$YNC= 000000        R$$JP0= 000000        T$$TRW= 000000
CTBS    000700R     002 C.SECE  001216RG  003 D$$YNM= 000000        R$$JP1= 000001        T$$VBF= 000000
```

```
                                                                L 14
SCP322 SCIP VERSION 3-22        MACRO M1104  06-DEC-77 08:28  PAGE 24-4
SYMBOL TABLE

T.ACTL  000052          T.OFF   000040          T2.AST= 100000          T3.ACP= 100000          UIBEF = 000001
T.ASTL  000016          T.PCB   000046          T2.BFX= 004000          T3.CAL= 000100          UIBSRH= 000000
T.ATT   000054          T.PRI   000002          T2.CAF= 000400          T3.CLI= 001000          U$$MHI= 000040
T.CPCB  000004          T.RCVL  000012          T2.CHK= 020000          T3.MCR= 004000          U$$MLO= 120000
T.DPPI  000040          T.RRFL  000064          T2.CKD= 010000          T3.NET= 000020          U$$MRN= 170324
T.EFLG  000022 G        T.SRCT  000063          T2.DST= 040000          T3.NSD= 000200          V$$CTR= 000400
T.IOC   000003          T.STAT  000032          T2.FXD= 002000          T3.PMD= 040000          X     = 000040
T.LBN   000041          T.ST2   000034          T2.HLT= 000200          T3.PRV= 010000          X$$DBT= 000000
T.LDV   000044          T.ST3   000036          T2.SPN= 000004          T3.REM= 020000          $RQSCH= ****** G
T.LNK   000000          T.TCBL  000030          T2.STP= 000020          T3.ROV= 000040          $SIGFL= ****** G
T.MXSZ  000050          T.UCB   000026          T2.TIO= 001000          T3.RST= 000400          $TKTCB= ****** G
T.NAM   000006          T2.ABO= 000100          T2.WFR= 000001          T3.SLV= 002000

  ABS.  000070    000
        000000    001
ISR     002216    002
XXX     001220    003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  3448 WORDS  ( 14 PAGES)
DYNAMIC MEMORY:  3866 WORDS  ( 14 PAGES)
ELAPSED TIME:  00:01:03
 256,020 SCIP:27, 256,030 SCIP/CRF/-SP=RSXMC,EXEMC/ML,RSX11M,OPDEF,SCIP
```

SCIP CREATED BY MACRO ON 6-DEC-77 AT 08:29 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL		VALUE		REFERENCES					
ASTEX	=	000563		19-953	#23-1282				
A$$CHK	=	000000		#1-35					
A$$CPS	=	000000		#1-29	#1-47				
A$$NSI	=	000000		#1-48					
A$$PRI	=	000000		#1-57					
A$$TRP	=	000000		#1-49	1-66				
BIPRES	=	000001		#2-7	8-391	8-394	8-411		
BIT00	=	000001		#5-206					
BIT01	=	000002		#5-207					
BIT02	=	000004		#5-208					
BIT03	=	000010		#5-209					
BIT04	=	000020		#5-210					
BIT05	=	000040		#5-211					
BIT06	=	000100		#5-212					
BIT07	=	000200		#5-213					
BIT08	=	000400		#5-214					
BIT09	=	001000		#5-215					
BIT10	=	002000		#5-216					
BIT11	=	004000		#5-217					
BIT12	=	010000		#5-218					
BIT13	=	020000		#5-219					
BIT14	=	040000		#5-220					
BIT15	=	100000		#5-221					
CAFTBL	=	000000		7-297	#7-322				
CBFLEN	=	000012		8-397	#8-403				
CBFTBL		000046	R	8-353	#8-397	8-403	15-734		
CBSIBL	=	000006		8-405	#8-409				
CBSIBT		000062	R	8-367	#8-405	8-409	12-509		
CCC	=	000257		#3-10					
CCFTBL	=	000000		7-304	#7-324				
CFKLU		000134	RG	12-496	#12-498				
CITA		001046	R	#24-1371	24-1377				
CITAA		000774	R	#24-1342	24-1346				
CITB		000752	R	16-768	16-775	#24-1328			
CITD		001124	R	#24-1400	24-1402				
CITE		001146	R	#24-1413	24-1415				
CITG		001176	R	#24-1431	24-1434				
CKW11	=	000000		#2-8					
CLALL	=	000257		#5-201	14-658	14-674	16-769	16-818	16-825
CLC	=	000241		#3-6	5-201	5-203			
CLN	=	000250		#3-9	5-201	5-202			
CLV	=	000242		#3-7	5-201	5-203			
CLVC	=	000243		#5-203	15-737				
CLZ	=	000244		#3-8	5-201	5-202			
CLZN	=	000254		#5-202	19-1021	19-1031	21-1171		
CMFTBL	=	000000		7-300	#7-323				
CMKTM	=	000433		#23-1283	24-1353				
CPFLEN	=	000014		7-332	#7-339				
CPFTBL		000012	R	7-308	#7-332	7-339	15-719		
CP.INI	=	******	G	7-331	7-336				
CP.KLL	=	******	G	7-331	7-338				
CP.MDM	=	******	G	7-331	7-334				

SYMBOL		VALUE		REFERENCES								
CP.MOD	=	******	G	7-331	7-335							
CP.OSU	=	******	G	7-331	7-333							
CP.RST	=	******	G	7-331	7-337							
CSPSND		000072	R	#10-431								
CTBS		000700	R	#23-1239	*24-1366							
CTDEL		000526	RG	17-850	21-1102	#21-1103						
CTDEL1		000550	R	#21-1138	21-1153							
CTDEL2		000570	R	21-1139	#21-1151							
CTDRET		000602	R	#21-1155								
CTEB		001204	R	21-1152	#23-1243	24-1345						
CTEMT		000350	RG	17-830	#17-838							
CTFTBL	=	000000		8-350	#8-387							
CTGVAC		000616	R	19-978	#22-1189							
CTGVEX		000642	R	22-1197	#22-1199							
CTIA		000514	R	19-980	#19-1030							
CTIB		000510	R	19-1015	#19-1023							
CTIME		000422	RG	17-854	19-976	#19-977						
CTLB		000704	R	21-1137	#23-1241	24-1341	24-1364					
CTOCK		000402	R	#19-872	23-1299							
CTPLR		000402	RG	18-860	#18-862							
CTPVAC		000650	R	19-905	19-962	19-1028	#22-1210					
CTQUE		000376	R	17-847	#17-853							
CTRET		001216	RG	6-226	#24-1453							
CTRETX		000474	R	19-917	#19-950	19-963						
CTRETZ		000512	R	19-894	#19-961							
CTSIBT	=	000000		8-364	#8-388							
CTSV		000702	R	22-1196	*22-1198	22-1217	*22-1218	#23-1240	*24-1365			
CTTAP		000466	R	19-907	#19-920							
CUTB	=	000030		#2-14	17-829	19-872	19-941	19-965	23-1242	23-1242	23-1242	23-1242
				23-1242	23-1242	23-1242	23-1242	24-1329	24-1363			
CUTNPS	=	000000		#2-15	24-1382							
C$$CKP	=	000000		#1-32	1-65	#1-95						
C$$ORE	=	004000		#1-96								
C$$RSH	=	177564		#1-87								
C.BARB	=	******	G	8-395	8-401							
C.BINT	=	******	G	8-395	8-402							
C.BLTL	=	******	G	8-383	8-395							
C.BMSE	=	******	G	8-396	8-407							
C.BODE	=	******	G	8-396	8-408							
C.BPD	=	******	G	8-395	8-398							
C.BPTC	=	******	G	8-395	8-400							
C.MKAS		000532	R	*19-999	#23-1289							
C.MKTM		000522	R	19-1000	#23-1285							
C.MKUN		000526	R	*19-993	#23-1287							
C.SAR		000000	R	#7-296	7-311	24-1426						
C.SARJ		000406	R	15-726	#15-739							
C.SARL	=	000010		7-296	#7-311							
C.SARP		000336	RG	6-224	14-698	#15-717	16-815	24-1431				
C.SARR		001216	RG	6-224	#24-1452							
C.SART		000350	R	#15-720	15-735							
C.SAT		000030	R	#8-349	8-356							
C.SATL	=	000004		8-349	#8-356							

SYMBOL	VALUE		REFERENCES							
C.SATN	000402	R	#15-737							
C.SATP	000366	RG	6-224	14-693	#15-732	16-811				
C.SBCY	000542	R	13-539	*13-548	14-633	*14-639	#23-1306	*24-1398		
C.SBEY	000742	R	13-545	14-668	#23-1318	24-1401				
C.SBGY	000072	R	#14-658	14-665						
C.SBPY	000546	R	13-538	*13-547	*13-590	#23-1311	*24-1395			
C.SBSY	000552	R	13-590	14-670	23-1311	23-1312	#23-1315	24-1394		
C.SECA	001216	RG	6-227	#24-1454						
C.SECB	001216	RG	6-227	#24-1455						
C.SECD	001216	RG	6-227	#24-1456						
C.SECE	001216	RG	6-227	#24-1457						
C.SECF	001216	RG	6-227	#24-1458						
C.SECG	001216	RG	6-227	#24-1459						
C.SECH	001216	RG	6-227	#24-1460						
C.SFCY	000134	R	14-634	#14-674						
C.SFEY	000550	R	14-635	*14-652	#23-1312	*24-1396				
C.SFGY	000254	R	#13-548	13-591						
C.SFJT	000036	R	#8-364							
C.SFLY	000132	R	14-669	#14-671						
C.SFQY	000120	R	14-651	#14-668						
C.SFVY	000304	R	13-546	#13-590						
C.SFY	000172	RG	6-224	#13-524	19-915					
C.SGAY	000314	R	13-526	#13-592						
C.SGBY	000322	R	13-540	#13-594						
C.SGET	000000	RG	14-631	#14-633						
C.SGEY	000260	R	#13-549	13-596						
C.SGMY	000076	R	14-637	#14-662						
C.SHI	000540	R	13-525	*13-594	14-636	*14-638	*14-662	*14-663	#23-1305	*24-1397
C.SLEY	000544	R	*13-592	*13-595	#23-1307	*24-1399				
C.SLTL	000042	R	#8-380	16-807						
C.SMAX	= 070000		#2-9							
C.SPS	000072	RG	6-224	#10-430						
C.SPTA	000166	R	14-689	#14-697						
C.SPUT	000142	RG	14-684	#14-685						
C.SREG	000074	RG	6-225	#11-443	12-498	15-718	15-733	19-882	19-977	21-1103
C.SRES	000116	RG	6-225	#11-459	12-511	15-744	19-950	19-1020	19-1030	21-1170
C.STAB	001214	RG	6-226	12-503	14-687	*16-801	*16-824	19-920	#23-1263	24-1406
C.STAE	001206	R	#23-1260							
C.STAK	001210	R	16-777	*16-809	#23-1261	*24-1408				
C.STAP	002216	R	23-1260	#23-1274	24-1414					
C.STAR	001216	RG	23-1271	#23-1272	24-1408					
C.STAT	= 177776		#5-186	13-535	13-536	13-552	14-644	14-645	14-656	15-742
			21-1114	21-1115	21-1168	22-1193	22-1194	22-1202	22-1214	22-1215
			22-1222	24-1338	24-1339	24-1350				
C.STIA	000212	R	16-767	#16-773						
C.STIB	000224	R	16-774	#16-776						
C.STIC	000236	R	16-778	#16-801						
C.STID	000332	R	#16-820							
C.STIF	000336	R	16-799	#16-823						
C.STIG	000342	R	16-821	#16-825						
C.STIH	000330	R	#16-819							
C.STII	000336	R	16-812	16-816	#16-824					

SCIP CREATED BY MACRO ON 6-DEC-77 AT 08:29 PAGE 4
SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE	REFERENCES
C.STIL	000334 R	#16-822
C.STIN	000176 RG	16-762 #16-763
C.STRT	001212 R	16-773 *16-817 #23-1262 *24-1328 24-1428
C.TLTL	= 000000	8-380 #8-389
C.USFY	= 000020	#2-11 13-524 13-530 13-539 13-569 13-589 23-1301 23-1316 23-1316 24-1393
C.USNL	= 000001	#2-16 23-1264 23-1264
C.USP	= 000300	6-236
C.USRT	= 000400	#2-17 23-1266 23-1272 23-1272 24-1407 24-1410
DPFRES	= 000001	#2-6 7-327 7-330 7-341
D$$BUG	= 177514	#1-88
D$$H11	= 000002	#1-126
D$$IAG	= 000000	#1-50
D$$ISK	= 000000	#1-33
D$$L11	= 000001	#1-125
D$$M11	= 000001	#1-127
D$$SHF	= 000000	#1-39
D$$YNC	= 000000	#1-34
D$$YNM	= 000000	#1-38
EMTVEC	= 000000	#2-10 6-243
E$$DVC	= 000000	#1-90 1-203 1-338
E$$NSI	= 000000	#1-91
E$$PER	= 000000	#1-92
E$$XPR	= 000000	#1-58
FAKELU	= 000001	#2-23 12-477 12-480
F$$LPP	= ******	1-373
F$$LVL	= 000001	#1-31
G$$TPP	= 000000	#1-54
G$$TTK	= 000000	#1-55
HALT	= 000000	#3-4
ISRMAX	= 000003	#2-19 7-298
I$$RAR	= 000000	#1-61
I$$RDN	= 000000	#1-36
JUNK	000534 R	19-990 19-991 #23-1299
K$$CNT	= 172542	#1-102
K$$CSR	= 172540	#1-101
K$$LDC	= 000024	#1-104
K$$TPS	= 000764	#1-103
LD$DB	= 000000	#1-106
LD$DK	= 000000	#1-109
LD$CS	= 000000	#1-112
LD$DT	= 000000	#1-114
LD$LP	= 000000	#1-116
LD$MM	= 000000	#1-119
LD$PP	= 000000	#1-121
LD$PR	= 000000	#1-122
LS111	= ******	13-531 13-549 14-640 14-653 15-739 21-1110 21-1165 22-1189 22-1199 22-1210 22-1219 24-1334 24-1347
L$$ASG	= 000000	#1-62
L$$DRV	= 000000	#1-51
L$$P11	= 000001	#1-118
L$$S11	= ******	1-311

SYMBOL	VALUE	REFERENCES							
L$$11R	= 000000	#1-117							
M$$CRB	= 000124	#1-135							
M$$CRX	= 000000	#1-131							
M$$EXT	= 000000	#1-99							
M$$FCS	= 000000	#1-30							
M$$MGE	= 000000	#1-37							
M$$MUP	= 000000	#1-43							
M$$OVR	= 000000	#1-130							
NOP	= 000240	#3-5	#5-197						
N$$LDV	= 000001	#1-129							
N$$MOV	= 000021	#1-60							
OFFSET	= 000000	#6-269							
PLF	= 000000	#2-26							
P$$D70	= 000000	#1-100	#1-105						
P$$GMX	= 000000	#1-42							
P$$LAS	= 000000	#1-40							
P$$P11	= 000001	#1-123							
P$$P45	= 000000	#1-97							
P$$RFL	= 000000	#1-53							
P$$RTY	= 000000	#1-93							
P$$R11	= 000001	#1-124							
P$$SRF	= 000000	#1-41							
Q$$OPT	= 000017	#1-59							
RSTS	= ******	11-443	11-459	12-506					
RSX11	= 000001	#2-22	7-288	13-600	15-702	15-748	19-972	19-1034	22-1180
R$$DER	= 000000	#1-86							
R$$EIS	= ******	1-358							
R$$EXV	= 000000	#1-79							
R$$JP0	= 000000	#1-107							
R$$JP1	= 000001	#1-108							
R$$JS1	= 000001	#1-113							
R$$KWC	= 000000	#1-110							
R$$K11	= 000001	#1-111							
R$$NDC	= 000003	#1-80							
R$$NDH	= 000226	#1-81							
R$$NDL	= 000001	#1-82							
R$$SND	= 000000	#1-56	#1-94						
R$$11M	= 000000	#1-136							
SCC	= 000277	#3-15							
SEC	= 000261	#3-11	5-198	5-199					
SEN	= 000270	#3-14	5-198	5-199	5-200				
SETALL	= 000277	#5-198							
SETNZ	= 000274	#5-200	15-738						
SETNZC	= 000275	#5-199							
SEV	= 000262	#3-12	5-198						
SEZ	= 000264	#3-13	5-198	5-199	5-200				
SIBEF	= 000040	6-264							
SLOCK	= 000000	#2-12	21-1105	21-1109	21-1156	21-1164			
SWB	= 000004	#2-5	6-230	6-252	6-252	6-252	6-253	6-257	6-268	6-271
		10-422	10-431	11-472	13-554	13-558	13-563	15-751	16-763	16-779
		16-798	19-868	19-952	19-957	19-992	19-998	19-1003	19-1017	19-1024
		23-1227	23-1246	23-1251	23-1278	23-1292	24-1352	24-1357	24-1416	24-1436

SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES								
			24-1439	24-1442							
SWBC	= 000000		#6-252	10-434	19-873	19-883	19-889	19-922	19-933	19-942	19-949
			19-981	19-989	19-1023	20-1038	21-1119	21-1136	23-1232	23-1245	23-1277
			24-1333	24-1367							
S$$WPC	= 000036		#1-83								
S$$WPR	= 000005		#1-84								
S$$WRG	= 000000		#1-89								
S$$YDF	= ******		1-1	1-438							
S$$YSZ	= 024000		#1-98								
TAPMAX	= 000001		#2-18	8-351	8-365	8-381					
TASKAD	001204	RG	13-555	23-1248	#23-1249	*24-1437					
TIMEDQ	= 000001		17-842	17-845	21-1094						
TLOCK	= 000001		#2-13	21-1104	21-1155	21-1175					
T$$ACR	= 000000		#1-63								
T$$BTW	= 000000		#1-45	#1-64							
T$$BUF	= 000000		#1-65								
T$$CCA	= 000000		#1-66								
T$$CCO	= 000000		#1-67								
T$$CTR	= 000000		#1-68								
T$$C11	= 000001		#1-115								
T$$GMC	= 000000		#1-69								
T$$GTS	= 000000		#1-70								
T$$J16	= 000001		#1-120								
T$$KMG	= 000000		#1-52								
T$$LWC	= 000000		#1-71								
T$$MAN	= 006307		#1-128								
T$$RNE	= 000000		#1-46	#1-72							
T$$RPR	= 000000		#1-73								
T$$RST	= 000000		#1-44	#1-74							
T$$RUB	= 000000		#1-75								
T$$SYN	= 000000		#1-76								
T$$TRW	= 000000		#1-77								
T$$VBF	= 000000		#1-78								
T.EFLG	000022	G	6-258	*13-556							
UIBEF	= 000001		6-261	13-556							
UIBSRH	= 000000		#2-20	23-1308							
U$$MHI	= 000040		#1-133								
U$$MLO	= 120000		#1-134								
U$$MRN	= 170324		#1-132								
U$$CTR	= 000400		#1-137								
X	= 000040		#7-298	7-299	7-301	#7-301	7-303	7-305	#7-305	7-307	7-309
			#7-309	#8-351	8-352	8-354	#8-354	#8-365	8-366	8-368	#8-368
			#8-381	8-382	8-384	#8-384	#23-1316	23-1317	23-1317	23-1317	23-1317
X$$DBT	= 000000		#1-85								
$RQSCH	= ******	G	6-258								
$SIGFL	= ******	G	6-258								
$TKTCB	= ******	G	6-258	24-1437							

MACRO NAME	REFERENCES	
CALL	#1-144	
CALLR	#1-170	
CRASH	#1-178	
DIRSV$	#1-186	
DRSTS	#1-194	
GTUCB$	#1-290	
INTLB	#1-204	
INTSE$	#1-214	
INTSV$	#1-256	
MFPS	#1-312	
MTPS	#1-315	
RETURN	#1-349	
SAVNP	#1-329	
SCBLB	#1-339	
SOB	#1-359	#24-1377
SWSTK$	#1-369	
TCBOF$	#2-28	#2-29

```
WRITE   MACRO M1104  06-DEC-77 08:30  PAGE 1


                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\ TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM


                                ;
                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                        .TITLE  WRITE
                        .IDENT  /00005/
                ;
                ; COPYRIGHT (C)  1975, 1976, 1977
                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                ;
                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                ;
                ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                ;
                ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                ; DEC.
                ;
                ; VERSION 0
                ;
                ; ROBERT H. ROSENBAUM
                ;
                ; LAST MODIFICATION ON 29-MAY-75
                ;
                ; RSX WRITE RECORD ROUTINE
                ;
000000                  .PSECT  MAIN
                ;
                ; MACRO CALLS
                        .MCALL  PUT$S,FDOFF$
000000                  FDOFF$  DEF$L           ;DEFINE FCS OFFSETS LOCALLY
                ;
                ; ASCII DEFINITIONS
000014                  FF=14
                ;
                ; INDICES INTO CHANNEL TABLE
000002                  COLPNT=2
000004                  RCSIZ=4
                ;
                ;+
                ; INPUTS:
                ;       R0 = BUFFER ADDRESS
                ;       R1 = BYTE COUNT
                ;
                ; OUTPUTS:
                ;       WRITES A RECORD
                ;       DOES VERTICAL AND HORIZONTAL FORMS PROCESSING
                ;-
                        .IF DF  RSX
000000  005004         WRIT$:: CLR     R4      ;BYTE COUNT OF EXPANDED RECORD
000002  116267  000406' 000000G         MOVB    CHRL(R2),CHARL  ;NUMBER OF CARRIAGE CONTROL CHARACTERS
000010  010267  000000G                 MOV     R2,COLAD        ;GET CHANNEL NUMBER
```

```
                                                BEQ     20$             ;CHANNEL 0
                                                MOV     #1,COLAD
                                        10$:    DEC     R2
                                                BEQ     20$
                                                ASL     COLAD
                                                BR      10$
                                        20$:    MOV     SWRSUP,R5       ;GET CORRECT FDB
                                                MOV     (R5),R3         ;FDBOUO OR FDBOU1
                                                MOV     F.NRBD+2(R3),R2 ;BUFFER ADDRESS
                                                TST     F.FNB+N.FID(R3) ;IS FILE CLOSED?
                                                BNE     30$             ;NO
                                                JMP     110$            ;YES -***THIS CODE SHOULD BE SHRUNK
                                        30$:    DEC     R1              ;FINISHED?
                                                BMI     50$             ;YES, WRITE RECORD
                                        40$:    INC     R4              ;INCREMENT BYTE COUNT
                                                MOVB    (R0)+,R3        ;GET CHARACTER
                                                MOVB    R3,(R2)+
                                                TSTB    SWRSL           ;SELECTIVE RECEIVE ACTIVE?
                                                BEQ     44$             ;NO
                                                CMP     #CHANL2,SWRSUP  ;IS OUTPUT GOING TO THE PRINT
                                                                        ;FILE (SELECTIVE RECEIVE ONLY)?
                                                BEQ     45$             ;YES, DO TAB EXPANSION
                                                BR      30$             ;NO, DON'T DO TAP EXPANSION FOR PUNCH DATA
                                        44$:    TSTB    SWRBN           ;NO, BINARY DATA?
                                                BNE     30$             ;YES, DON'T DO TAB EXPANSION
                                        45$:    CMPB    R3,#11          ;IS IT A TAB?
                                                BNE     30$             ;NO
                                                TSTB    HFCACT          ;HORIZONTAL TAB ACTIVE
                                                BPL     30$             ;NO
                                                MOVB    #40,-1(R2)      ;REPLACE SPACE WITH TAB
                                                TSTB    HFCREC(R4)      ;TAB STOP
                                                BNE     30$             ;YES
                                                TSTB    -(R0)           ;NO
                                                BR      40$
                                        50$:    MOV     (R5),R0         ;GET FDB
                                                CMPB    #1,F.RTYP(R0)   ;FIXED LENTH RECORD
                                                BNE     60$             ;NO
                                                ADD     R4,RCSIZ(R5)    ;ADD INTO TOTAL RECORD SIZE
                                                MOV     R2,F.NRBD+2(R0) ;UPDATE BUFFER POINTER
                                                CMP     RCSIZ(R5),F.RSIZ(R0)
                                                BLO     120$
                                                MOV     RCSIZ(R5),R4    ;NUMBER OF CHARACTERS TO WRITE
                                                SUB     R4,F.NRBD+2(R0) ;REAJUST PNTR FOR FCS
                                                CLR     RCSIZ(R5)
                                        60$:    PUT$S   ,,R4
                                                BCS     110$            ;OH NO AN ERROR
                                        62$:    TSTB    SWRSL           ;SELECTIVE RECEIVE ACTIVE?
                                                BEQ     64$             ;NO
                                                CMP     #CHANL2,SWRSUP  ;IS OUTPUT GOING TO THE PRINT FILE
                                                BEQ     65$             ;YES, CHECK FOR CARRIAGE CONTROL
                                                BR      120$            ;NO, DON'T CHECK SINCE ITS PUNCH DATA
                                        64$:    TSTB    SWRBN           ;BINARY DATA?
                                                BNE     120$            ;YES, IGNORE CARRIAGE CONTROL
                                        65$:    CMP     COLAD,#1        ;SKIP TO CHANNEL 1
                                                BNE     70$             ;NO
                                                MOV     F.NRBD+2(R0),R2 ;GET BUFFER ADDRESS
                                                MOVB    #FF,(R2)+
```

```
    115 000304                                          PUTSS   ,,#1             ;WRITE RECORED
    116 000316  103424                                  BCS     110$            ;ERROR
    117 000320  000407                                  BR      80$
    118 000322                                  70$:
    119 000322  062765  000002  000002                  ADD     #2,COLPNT(R5)   ;INCREMENT COLUMN COUNT
    120 000330  026527  000002  000204G                 CMP     COLPNT(R5),#COLTT+132.  ;A WHOLE FORM
    121 000336  103403                                  BLO     90$             ;NO
    122 000340  012765  000000G 000002  80$:   MOV     #COLTT,COLPNT(R5)
    123 000346  036775  000000G 000002  90$:   BIT     COLAD,@COLPNT(R5)       ;A STOP?
    124 000354  001003                                  BNE     100$            ;YES
    125 000356  105367  000000G                         DECB    CHARL           ;NOT DOUBLE OR TRIPLE SPAC?
    126 000362  001407                                  BEQ     120$            ;NO, EXIT
    127 000364  005004                          100$:   CLR     R4              ;YES, WRITE NULL BLOCK
    128 000366  000673                                  BR      50$
    129 000370  152767  000200  000000G 110$:   BISB    #200,IOERR      ;AN ERROR
    130 000376  000261                                  SEC                     ;RETURN WITH C BIT SET
    131 000400  000401                                  BR      130$
    132 000402  000241                          120$:   CLC                     ;GOOD RETURN - CLEAR CARRY
    133 000404  000207                          130$:   RTS     PC
    134                                                 .ENDC
```

```
                                        ;+
                                        ; CARRIAGE CONTROL TABLE FOR RSX
                                        ;-
000406      001                 CHRL:   .BYTE   1       ;CHANNEL 0
000407      001                         .BYTE   1       ;CHANNEL 1
000410      001                         .BYTE   1       ;CHANNEL 2
000411      001                         .BYTE   1       ;CHANNEL 3
000412      001                         .BYTE   1       ;CHANNEL 4
000413      001                         .BYTE   1       ;CHANNEL 5
000414      001                         .BYTE   1       ;CHANNEL 6
000415      001                         .BYTE   1       ;CHANNEL 7
000416      001                         .BYTE   1       ;CHANNEL 8
000417      002                         .BYTE   2
000420      003                         .BYTE   3
                                        .EVEN
          000001                        .END
```

```
WRITE   MACRO M1104  06-DEC-77 08:30  PAGE 3-1
SYMBOL TABLE

CHANL2= ****** GX        F.CHR  = 000075        F.MBC1= 000055        F.VBN  = 000064        RSX    = 000001
CHARL  = ****** GX        F.CNTG= 000034        F.MBFG= 000056        F.VBSZ= 000060        RSX11M= 000001
CHPL     000406R      002 F.DFNB= 000046        F.NRBD= 000024        HFCACT= ****** GX     SWRBN  = ****** GX
COLAD  = ****** GX        F.DSPT= 000044        F.NREC= 000030        HFCREC= ****** GX     SWRSL  = ****** GX
COLPNT= 000002            F.DVNM= 000134        F.OVBS= 000030        IOERR  = ****** GX     SWRSUP= ****** GX
COLTT  = ****** GX        F.EFBK= 000010        F.RACC= 000016        NRCVSF= 000000        S.FATT= 000016
FF     = 000014           F.EFN  = 000050        F.RATT= 000001        NTRNSF= 000002        S.FDB  = 000140
F.ACTL= 000076            F.EOBB= 000032        F.RCNM= 000034        N.DID  = 000024        S.FNAM= 000006
F.ALOC= 000040            F.ERR  = 000052        F.RCTL= 000017        N.DVNM= 000032        S.FNB  = 000036
F.BBFS= 000062            F.FACC= 000043        F.RSIZ= 000002        N.FID  = 000000        S.FNBW= 000017
F.BDB  = 000070           F.FFBY= 000014        F.RTYP= 000000        N.FNAM= 000006        S.FNTY= 000004
F.BGBC= 000057            F.FNAM= 000110        F.SEQN= 000100        N.FTYP= 000014        S.FTYP= 000002
F.BKDN= 000026            F.FNB  = 000102        F.SPDV= 000072        N.FVER= 000016        S.NFEN= 000020
F.BKDS= 000020            F.FTYP= 000116        F.SPUN= 000074        N.NEXT= 000022        TAGNUM= 000000
F.BKEF= 000050            F.FVER= 000120        F.STBK= 000036        N.STAT= 000020        WRITS    000000RG     002
F.BKP1= 000051            F.HIBK= 000004        F.UNIT= 000136        N.UNIT= 000034        .PUTSQ= ****** G
F.BKST= 000024            F.LUN  = 000042        F.URBD= 000020        RCSIZ  = 000004        ...TPC= 000140
F.BKVB= 000064            F.MBCT= 000054

.  ABS.   000000      000
          000000      001
MAIN     000422      002
ERRORS DETECTED:   0

VIRTUAL MEMORY USED:   1817 WORDS  ( 8 PAGES)
DYNAMIC MEMORY:   2810 WORDS  ( 10 PAGES)
ELAPSED TIME:   00:00:19
 256,020 WRITE,27, 256,030 WRITE/CRF/-SP=UAPRXM,WRITE
```

```
WRITE       CREATED BY  MACRO  ON 6-DEC-77 AT 08:30      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF
SYMBOL  VALUE              REFERENCES
CHANL2  = ******  GX       2-77        2-106
CHARL   = ******  GX      *2-56       *2-125
CHRL      000406 R         2-56       *3-139
COLAD   = ******  GX      *2-57       *2-59       *2-62       2-111       2-123
COLPNT  = 000002          *2-42       *2-119       2-120      *2-122       2-123
COLTT   = ******  GX       2-120       2-122
FF      = 000014          *2-39        2-114
F FNB   = 000102           2-67
F NRBD  = 000024           2-66       *2-96       *2-100      *2-102       2-113      *2-115
F RSIZ  = 000002           2-97
F RTYP  = 000000           2-93
HFCACT  = ******  GX       2-85
HFCREC  = ******  GX       2-88
IOERR   = ******  GX      *2-129
NRCVSF  = 000000          *1-5
NTRNSF  = 000002          *1-4
N FID   = 000000           2-67
RCSIZ   = 000004          *2-43       *2-95        2-97        2-99       *2-101
RSX     = 000001          *1-2         2-54
RSX11M  = 000001          *1-3
SWRBN   = ******  GX       2-81        2-109
SWRSL   = ******  GX       2-75        2-104
SWRSUP  = ******  GX       2-64        2-77        2-106
TAGNUM  = 000000          *1-8
WRITS     000000 RG       *2-55
.PUTSQ  = ******  G        2-102       2-102       2-115       2-115
```

MACRO CROSS REFERENCE CREF

MACRO NAME	REFERENCES					
CMOVSW	#2-102	2-102	2-102	#2-115	2-115	2-115
ERROR	#1-9					
ERRORK	#1-16					
ERRORM	#1-32					
ERRORP	#1-47					
ERRS	#2-102	#2-102	#2-115	#2-115		
FATAL	#1-53					
FDOFFS	#2-35	#2-36				
LDFDBS	#2-102	2-102	#2-115	2-115		
PUTSS	#2-35	2-102	2-115			

COMTEX-MACROS MACRO M1104 06-DEC-77 08:29
TABLE OF CONTENTS

3- 21 COMTEX MACRO DEFINITIONS
5- 57 COMMUNICATIONS OUTPUT DONE
6- 152 OPEN FILE FOR TRANSMIT
8- 221 READ
9- 245 READ COMPLETE
10- 286 BISYNC OUTPUT
11- 324 PARAMETER BLOCKS
12- 338 CONSTANT DATA AREA

```
TRANSMIT        MACRO M1104  06-DEC-77 08:29  PAGE 1


                                ;PREFIX FILE FOR 2780 UAP RUNNING UNDER RSX11M
000001                          RSX=1
000001                          RSX11M=1
000002                          NTRNSF=2                ;NUMBER OF FILES - 1 TO TRANSFER IN ONE CMMD STRING (3)
000000                          NRCVSF=0                ;NUMBER OF FILES - 1 TO RECEIVE (1)
                                ;
                                ;MACRO DEFINITIONS
000000                                  TAGNUM=0
                                        .MACRO  ERROR   TEXT
                                        ERRORK  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM
                                ;
                                ; MACRO TO DO A KILL AND QUEUE THE MESSAGE
                                ;
                                        .MACRO  ERRORK  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMSG
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ;
                                ; MACRO TO QUEUE A MESSAGE ONLY
                                ;
                                        .MACRO  ERRORM  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        JSR     R5,TYPMS1
                                        .WORD   TAG'N
                                        .WORD   TAGE'N-TAG'N
                                        .PSECT  MESAG2,GBL
                                TAG'N:  .ASCII  "TEXT"
                                TAGE'N:
                                        .PSECT  MAIN
                                        .ENDM
                                        .ENDM

                                ; MACRO TO QUEUE A MESSAGE AND PROMPT
                                ;
                                        .MACRO  ERRORP  TEXT
                                        ERRORM  <TEXT>
                                        JSR     PC,PROMPT
                                        .ENDM


                                        .MACRO  FATAL   TEXT
                                        ERRORK  <TEXT>
                                        JSR     R5,DIE
                                        .ENDM
```

```
                        .TITLE  COMTEX-MACROS
                        .IDENT  /00002/
                ;
                ;COPYRIGHT 1974, 1975, 1976, 1977
                ;DIGITAL EQUIPMENT CORP.
                ;146 MAIN STREET          MAYNARD, MASS. 01754
                ;
                ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR
                ;RELIABILITY OF ITS SOFTWARE ON EQUIPMENT WHICH
                ;IS NOT SUPPLIED BY DIGITAL
                ;
                ; VERSION 0
                ;
                ; 23-SEPTEMBER-74
                ;
                ;VERSION 1
                ; SEP 1977 - CEP
                ; MODIFICATIONS FOR CUT CODE VERSION OF 2780
                ; LIBRARY OF COMTEX MACRO DEFINITIONS
```

```
COMTEX-MACROS   MACRO M1104  06-DEC-77 08:29  PAGE 3
COMTEX MACRO DEFINITIONS

                        .SBTTL  COMTEX MACRO DEFINITIONS
                ;
                        .MACRO  LINIT,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTINT
                        .ENDM
                ;
                        .MACRO  ASRBUF,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTPUT
                        .ENDM
                ;
                        .MACRO  PUTTC,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTPUT
                        .ENDM
                ;
                        .MACRO  PUTMC,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTPUT
                        .ENDM
                ;
                        .MACRO  GETS
                        JSR     PC.EMTGET
                        .ENDM
                ;
                        .MACRO  PUTTM,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTTIM
                        .ENDM

                        .MACRO  PUTD,PARLST
                        MOV     #PARLST,R1
                        JSR     PC,EMTPUT
                        .ENDM
```

```
                        .TITLE  TRANSMIT
                        .IDENT  /00017/
                ;
                ; COPYRIGHT (C) 1974,1975,1976,1977
                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                ;
                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY
                ; ON A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                ; SOFTWARE, OR ANY COPIES THEREOF, MAY NOT BE PROVIDED OR
                ; OTHERWISE MADE AVBAILABLE TO ANY OTHER PERSON EXCEPT
                ; FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO THESE
                ; LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE SOFTWARE
                ; SHALL AT ALL TIMES REMAIN IN DEC.
                ;
                ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                ;
                ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                ; DEC.
                ;
                ; VERSION 0
                ;
                ; ROBERT H. ROSENBAUM   20-SEPTEMBER-74
                ;
                ; B.L. RUBENSTEIN       27-JAN-75
                ; MODIFIED FOR RT-11  V02-1  (F/B)
                ; LAST MODIFICATION ON 28-OCT-75
                ; MODIFIED FOR CUT CODE VERSION
                ; CEP 8/77
                ;
                ;+
                ; TRANSMIT DATA COMPLETE ROUTINE
                ;
                ; CALL:
                ;       MSGDON IS CALLED F    THE "GETS" LOOP THROUGH THE JUMP TABLE
                ;       UPON FINDING A U16    TRY FROM THE BISYNC TAP.
                ;
                ; FUNCTION:
                ;       THIS ROUTINE READS THE FILE BEING TRANSMITTED AND HANDLES
                ;       ALL THE BUFFER MANIPULATION AND CORRDINATION WITH THE TAP.
                ;-
                        .IF DF RSX
000000                  .PSECT  MAIN
                        .ENDC
                        .IF DF RT11
                        .CSECT  TRNSMT
                ;
                ; MACRO CALLS
                        .MCALL  .PRINT,,V2,,
                        ,,V2,,
                        .ENDC
```

```
TRANSMIT        MACRO M1104  06-DEC-77 08:29  PAGE 5
COMTEX MACRO DEFINITIONS

                                        .SBTTL  COMMUNICATIONS OUTPUT DONE
                                        .GLOBL  MSGDON,INITF,ENTRY,FLAGS,EOTPEN,EOTFLG
                                        .GLOBL  OFFHAN,TVAB,TDONE,ELATCH
                                        .GLOBL  GETENT,ENDPEN,RVIPEN,IOERR
                                        .GLOBL  MODE,CMPAR,TXPEN,ITBM,BINEOT,CLOSER
                                        .GLOBL  EOTSEQ,SWRBN,TURNAR,ENBPAR
                                        .GLOBL  CODE,QTRAN,QMODE,CRBFAV,TRNSBLK
                                        .GLOBL  RED$,STOREX,CMPLST,CRBASX,CMCLST
                                        .GLOBL  RECSIZ,CMBUFE,CMFLST,NXTFIL
                                        .GLOBL  SWREN,FSTGET,CMBMAX,CMBFPN,PRINTX
                                        .GLOBL  BASELN,XSTEQ,XFEREN,CMBUF1,XMTERR,RCVBLK
                                        .GLOBL  CLOSET,MRKFIL
                                        .IF DF RT11
                                        .GLOBL  RTEND
                                        .ENDC

000000  012705  000006G        MSGDON: MOV     #ENTRY+6,R5
000004  105767  000000G                TSTB    INITF           ;INITIALIZATION?
000010  100545                         BMI     MASTD           ;YES
000012  105767  000000G                TSTB    FLAGS           ;MASTER OR SLAVE
000016  100040                         BPL     GOG1            ;SLAVE, IGNORE ANY DONES
000020  106367  000000G                ASLB    TVAB            ;ABORT?
000024  103433                         BCS     GOJMP2          ;YES
000026  061567  000000G                ADD     (R5),XMTERR
000032  005745                         TST     -(R5)           ;CHECK FOR ERROR
000034  100425                         BMI     GOJMP1          ;ERROR PRESENT
000036  106367  000000G                ASLB    EOTPEN          ;EOT PENDING FLAG SET?
000042  103030                         BCC     MSGTA2          ;NO
000044  106367  000000G                ASLB    EOTFLG          ;YES, EOT FLAG SET?
000050  103002                         BCC     GOJMP           ;NO
000052  000167  000000G                JMP     OFFHAN          ;YES, GO OFFLINE
000056  106367  000000G        GOJMP:  ASLB    TDONE           ;HAVE ALL FILES BEEN TRANSMITTED?
000062  103410                         BCS     ARND
000064  105067  000000G                CLRB    ELATCH
000070  105767  000000G                TSTB    RVIPEN          ;RVI PENDING?
000074  100403                         BMI     ARND            ;YES, DON'T SET TRANSMIT PENDING
000076  152767  000200  000000G        BISB    #200,TXPEN      ;SET TRANSMIT PENDING
000104  000167  000000G        ARND:   JMP     TURNAR

000110  000167  000412        GOJMP1: JMP     GOABT1  ;ABORT TRANS - SET EOT
000114  000167  000434        GOJMP2: JMP     GOABRT  ;ABORT TRANSMISSION - SET EOT

000120  000167  000000G        GOG1:   JMP     GETENT          ;GET NEXT UIB ENTRY
000124  032715  040000        MSGTA2: BIT     #40000,(R5)     ;RVI INDICATED?
000130  001414                         BEQ     MSGTC           ;NO, GO ON
000132  152767  000200  000000G        BISB    #200,ENDPEN     ;YES, SET END PENDING FLAG
000140  152767  000200  000000G        BISB    #200,RVIPEN     ;SET RVI PENDING
000146                                 ERROR   <RVI>
                                                               ;CONTROL MODE
                                       ;COMM BUFFER PROCESSED BY BISYNC TAP

000162  032715  000400        MSGTC:  BIT     #400,(R5)       ;HAS FTX RECORD BEEN ACK'ED?
000166  001422                         BEQ     MSGTD           ;NO - MORE RECORDS PENDING
000170  105767  000000G                TSTB    TDONE           ;TRANSMIT DONE?
000174  100401                         BMI     10$             ;YES
```

```
    113 000176  000414                              BR      NMES            ;SEND AN EOT
    114 000200  132767  000040  000000G 10$:        BITB    #40,TXPEN       ;BUT RVI STILL PENDING?
    115 000206  001010                              BNE     NMES            ;YES, THEN TRANSMIT NOT REALLY DONE
    116 000210  105067  000000G                     CLRB    RVIPEN          ;IF REALLY COMPLETED TRANMISSION THEN
    117                                                                     ;FORGET RVI
    118 000214                                      ERROR   <TRNS COMPLETE>
    119 000230                          NMES:
    120 000230  000167  000342                      JMP     TERT            ;TRANSMIT EOT
    121 000234  106367  000000G         MSGTD:      ASLB    IOERR           ;AN I/O ERROR ON READ?
    122 000240  103011                              BCC     MAXTD           ;NO
    123 000242                                      PUTD    STXEQ           ;SEND STX-ENQ
    124 000252  112767  000200  000000G             MOVB    #200,TVAB       ;FLAG ERROR CONDITION DETECTED
    125 000260  000167  000000G                     JMP     GETENT
    126 000264  105767  000000G         MAXTD:      TSTB    MODE            ;TRANSMIT ABORTED?
    127 000270  001713                              BEQ     GOG1            ;YES
    128 000272  005067  000004G                     CLR     CMPAR+4
    129 000276  000167  000736                      JMP     CMOUT           ;NO - CONTINUE TRANSMIT
    130
    131                                 ;BID CONTENTION - LET REMOTE STATION HAVE THE LINE BUT
    132                                 ; SIMULATE AN RVI CONDITION SO WE CAN TRY AGAIN AFTER
    133                                 ; THE RECEPTION
    134 000302  105067  000000G         COEND:      CLRB    TXPEN           ;CLEAR TRANSMIT PENDING
    135 000306  004767  000000G                     JSR     PC,CLOSET       ;CLOSE THE TRANSMIT FILE
    136 000312  152767  000200  000000G             BISB    #200,RVIPEN     ;SIMULATE AN RVI CONDTION
    137 000320  000167  000000G                     JMP     TURNAR          ;TURN THE LINE AROUND
    138
    139
    140
    141
    142 000324  105767  000000G         MASTD:      TSTB    FLAGS           ;MASTER OR SLAVE
    143 000330  100004                              BPL     MSTG1
    144 000332  005745                              TST     -(R5)           ;CHECK FOR VALID
    145                                                                     ;RESPONSE TO BID
    146 000334  100507                              BMI     GOABRT          ;INVALID, GO ABORT
    147 000336  105715                              TSTB    (R5)            ;TRANSMIT OVERRIDEN BY CONTENTION?
    148 000340  100760                              BMI     COEND           ;YES - GO INTO RECEIVE MODE
    149                                                                     ;NO - PROCEED NORMALLY
    150
```

```
TRANSMIT        MACRO M1104  06-DEC-77 08:29  PAGE 6
OPEN FILE FOR TRANSMIT

                                                .SBTTL  OPEN FILE FOR TRANSMIT
000342  152767  000020  000000G MSTG1:  BISB    #20,FLAGS
                                        .IF DF  RT11
                                        TSTB    SWRBN
                                        .ENDC
                                        .IF DF  RSX
000350  105767  000000G                 TSTB    FLAGS           ;MASTER?
000354  100404                          BMI     MMAS            ;YES
000356  105767  000000G                 TSTB    SWRBN           ;RECEIVE BINARY?
000362  001415                          BEQ     CHDFIX          ;NO
000364  000403                          BR      BINST           ;YES
000366  105767  000000G         MMAS:   TSTB    SWTBN           ;BINARY MODE?
                                        .ENDC
000372  001411                          BEQ     CHDFIX
000374  152767  000030  000000G BINST:  BISB    #30,FLAGS       ;YES, SET NATIVE MODE
000402  012767  000000G 000000G         MOV     #BINEOT,EOTSEQ
000410  012767  000000G 001050         MOV     #BINTTD,STEQM   ;SET FOR BINARY TTD SEQUENCE
000416  016767  000000G 000002G CHDFIX: MOV     XFEREN,ENBPAR+2 ;VALID, MOVE CONTROL WORD TO PARAM LIST
000424  116767  000000G 000002G         MOVB    FLAGS,ENBPAR+2  ;SET CURRENT OPTION SELECTIONS
000432  156767  000000G 000003G         BISB    CODE,ENBPAR+3   ;SET CODE SELECTION
000440                                  PUTTC   ENBPAR          ;HANDLE XFER ENABLE
000450  142767  000010  000000G         BICB    #10,FLAGS       ;CLEAR NATIVE MODE SELECT
000456  105067  000000G                 CLRB    INITF           ;CLEAR INIT FLAG
000462  105767  000000G                 TSTB    FLAGS           ;MASTER OR SLAVE?
000466  100402                          BMI     TRNC            ;MASTER
000470  000167  000000G                 JMP     GETENT          ;SLAVE
000474  112767  000002  000000G TRNC:   MOVB    #2,MODE         ;SET TRANSMIT MODE
000502  105067  000000G                 CLRB    TXPEN           ;X-MIT REQUEST SATISFIED
000506  016767  000756  000000G         MOV     QTRAN,QMODE     ;SET UP FOR QUERY TYPEOUT
000514  112767  000002  000000G         MOVB    #2,CRBFAV       ;INIT # OF CR BUFFERS
000522  000167  000216                  JMP     CRAGN

                                ;
                                        .GLOBL  GOABRT,TER1
                                ;
000526                          GOABT1:
000526  022765  000010  000002          CMP     #10,2(R5)       ;SEE IF ABORT CAUSED BY NAKS*****
000534  003007                          BGT     GOABRT
000536                                  ERROR   <NAK ABRT>
000552  000406                          BR      XXX
000554                          GOABRT:
000554                                  ERROR   <OPR ABORT>     ;TYPE ABORT MESSAGE
000570  152767  000200  000000G XXX:    BISB    #200,TDONE      ;TRANSMIT DONE
000576                          TERT:   PUTD    EOTPAR          ;GO SEND EOT
000606  100446                          BMI     NOTRNS          ;DON'T CLOSE FILE IF ERROR
000610  126727  000000G 000002          CMPB    MODE,#2         ;TRANSMIT MODE?
000616  001012                          BNE     TER2            ;NO
000620  004767  000000G         TER1:   JSR     PC,CLOSET       ;CLOSE THE TRANSMIT FILE
000624  105767  000000G                 TSTB    TDONE           ;TRANSMIT DONE?
000630  100016                          BPL     TOT
000632  105767  000000G                 TSTB    RVIPEN          ;IS RVI PENDING?
000636  100413                          BMI     TOT             ;YES, DON'T CLEAR TRANSMIT FILE NAME
                                                                ; SO THAT WE CAN REOPEN LATER
000640  005077  000000G                 CLR     @TRNSBLK
000644  126727  000000G 000001  TER2:   CMPB    MODE,#1         ;RECEIVE MODE?
000652  001005                          BNE     TOT             ;NO
000654  004767  000000G                 JSR     PC,CLOSER       ;CLOSE THE RECEIVE FILES
```

```
TRANSMIT        MACRO M1104  06-DEC-77 08:29  PAGE 6-1
OPEN FILE FOR TRANSMIT

    209 000660  012777  177777  000000G .IIF DF RSX     MOV     #-1,@RCVBLK     ;USE DEFAULT RECEIVE FILE
    210                                 .IIF DF RT11    CLR     @RCVBLK         ;USE DEFAULT RECEIVE FILE
    211 000666  152767  000200  000000G TOT:    BISB    #200,EOTPEND    ;SET EOT PEND FLAG
    212 000674  152767  000200  000000G        BISB    #200,FLAGS      ;SET FOR MASTER MODE
    213 000702  112767  000002  000000G        MOVB    #2,MODE         ;SET TO TRANSMIT MODE
    214 000710  116767  000000G 000000G        MOVB    PRINTX,STOREX   ;REALIGN INDEX POINTERS
    215 000716  105067  000000G                CLRB    INITF           ;DONT NEED AN INIT PHASE
    216 000722  000406                         BR      CRPGN
    217 000724                          NOTRNS: ERROR   <TER RJT>
    218 000740  000167  000000G         CRPGN:  JMP     GETENT          ;GO TO GET NEXT ENTRY
```

```
TRANSMIT        MACRO M1104  06-DEC-77 08:29  PAGE 8
READ

                                                .SBTTL  READ

000744  105767  000000G         CRAGN:  TSTB    CRBFAV          ;ANY BUFFERS FOR READ
000750  003773                          BLE     CRPGN
000752  105767  000000G                 TSTB    ELATCH          ;READ BLOCKED?
000756  100770                          BMI     CRPGN
000760  105767  000000G                 TSTB    RVIPEN          ;RVI PENDING?
000764  100002                          BPL     CRANW           ;NO
000766  004767  000000G                 JSR     PC,MRKFIL       ;YES, MARK PLACE IN FILE
000772  016700  000000G         CRANW:  MOV     CRBASX,R0       ;BUFFER ADDRESS
000776  004767  000000G                 JSR     PC,RED$         ;READ
001002  103516                          BCS     CMOUT           ;A READ ERROR
001004  116702  000000G         CRAOK:  MOVB    STOREX,R2
001010  016762  000000G 000000G         MOV     CRBASX,CMPLST(R2)       ;STORE BUFFER ADDRESS
001016  010162  000000G                 MOV     R1,CMCLST(R2)   ;AND WORD COUNT
001022  016700  000000G                 MOV     RECSIZ,R0
001026  005720                          TST     (R0)+
001030  060067  000000G                 ADD     R0,CRBASX
001034  066700  000000G                 ADD     CRBASX,R0
001040  020027  000000G                 CMP     R0,#CMBUFE
001044  101403                          BLOS    DFIDN
001046  012767  000000G 000000G         MOV     #CMBUF1,CRBASX
001054                          DFIDN:
```

```
                                                .SBTTL  READ COMPLETE
001054  005062  000000G          CARC:   CLR     CMFLST(R2)          ;WORD TO MARK EOF
                                        .IF DF RT11
                                        ASLB    RTEND               ;END OF FILE?
                                        BCC     CARC5               ;NO
                                        .ENDC
                                        .IF DF  RSX
001060  122767  000000G 000000C         CMPB    #IE.EOF,F.ERR+FDBIN     ;END OF FILE
001066  001036                         BNE     CARC5
                                        .ENDC
001070  004767  000000G                 JSR     PC,NXTFIL           ;ANOTHER FILE?
001074  001413                         BEQ     CARCE
001076  142767  000040  000000G        BICB    #40,TXPEN           ;TRANSMIT PENDING
001104  105767  000000G                TSTB    SWREN               ;/E SWITCH
001110  001016                         BNE     CARCB               ;YES
001112  004767  000000G                JSR     PC,FSTGET           ;DO THE INITIAL GET
001116  105162  000001G                COMB    CMFLST+1(R2)        ;FLAG THAT A MODE CHANGE IS NEEDED
001122  000420                         BR      CARC5
001124  005077  000000G          CARCE: CLR     @TRNSBLK            ;ABORT COMMAND
001130  105067  000000G                CLRB    RVIPEN              ;IF REALLY FINISHED FORGET ABOUT RVI
001134  152767  000200  000000G CARCA: BISB    #200,TDONE
001142  004767  000000G                JSR     PC,CLOSET           ;CLOSE THE TRANSMIT FILE
001146  112767  000200  000000G CARCB: MOVB    #200,ELATCH
001154  112762  000040  000000G        MOVB    #40,CMFLST(R2)      ;SET LAST BLOCK INDICATOR
001162  000413                         BR      CARC6
001164  106367  000000G          CARC5: ASLB    ENDPEN
001170  103761                         BCS     CARCA
001172  016203  000000G                MOV     CMPLST(R2),R3       ;GET BUFFER POINTER
001176  066203  000000G                ADD     CMCLST(R2),R3       ;FIND END OF RECORD
001202  116713  000000G                MOVB    ITBM,(R3)           ;LOAD ITB
001206  005262  000000G                INC     CMCLST(R2)
001212  005722                 CARC6: TST     (R2)+               ;INCREMENT QUEUE LOAD INDEX
001214  020227  000000G                CMP     R2,#CMBMAX          ;AT END OF QUEUE
001220  002401                         BLT     CARC7               ;NO
001222  005002                         CLR     R2                  ;YES - RESET TO BEGINNING

001224  110267  000000G          CARC7: MOVB    R2,STOREX           ;RESTORE UPDATED INDEX
001230  105267  000000G                INCB    CMBFPN              ;ANOTHER BUFFER PENDING FOR OUTPUT
001234  105367  000000G                DECB    CRBFAV
```

```
                                        .SBTTL  BISYNC OUTPUT
                                        .GLOBL  XEOTS,EOTSEQ,BINEOT,STEQM,ENBPAR,BINTTD
001240  005767  000004G        CMOUT:  TST     CMPAR+4         ;PUTD TO BISYNC ACTIVE?
001244  001100                        BNE     CMOUTX
001246  105767  000000G                TSTB    CMBFPN
001252  003475                        BLE     CMOUTX
001254  116700  000000G                MOVB    PRINTX,R0       ;GET QUEUE UNLOAD INDEX
001260  016067  000000G 000002G        MOV     CMPLST(R0),CMPAR+2      ;GET BUFFER ADDRESS
001266  016067  000000G 000004G        MOV     CMCLST(R0),CMPAR+4      ;AND BYTE COUNT
001274  116067  000000G 000001G        MOVB    CMFLST(R0),CMPAR+1
001302                          CMOUT1:
                                        ;GIVE BUFFER TO BISYNC TAP
001302                                  PUTD    CMPAR
001312  100455                        BMI     CMOUTX
                                                                ;AGAIN AFTER X-MIT DONE
001314  105760  000001G                TSTB    CMFLST+1(R0)    ;IS MODE CHANGE NEEDED?
001320  001437                        BEQ     20$
001322  012767  000000G 000000G        MOV     #XEOTS,EOTSEQ   ;SET FOR NORMAL EOT SEQUENCE
001330  012767  000000G 000130        MOV     #XSTEQ,STEQM    ;AND NORMAL TTD SEQUENCE
001336  105767  000000G          .IIF DF RSX   TSTB    SWTBN   ;BINARY TRANSMISSION REQUESTED?
                                 .IIF DF RT11  TSTB    SWBEN           ;BINARY TRANSMISSION REQUESTED?
001342  001411                        BEQ     10$             ;NO. ALL OK
001344  012767  000000G 000000G        MOV     #BINEOT,EOTSEQ  ;YES UEE BINARY EOT
001352  012767  000000G 000106        MOV     #BINTTD,STEQM   ;AND BINARY TTD SEQUENCE
001360  152767  000030  000000G        BISB    #30,FLAGS       ;USE NATIVE MODE
001366  116767  000000G 000036G 10$:   MOVB    CODE,ENBPAR+3   ;USE STATUS OPTION
001374  116767  000000G 000002G        MOVB    FLAGS,ENBPAR+2  ;SET CURRENT OPTIONS SETTINGS
001402                                  PUTTC   ENBPAR          ;CALL TAP
001412  142767  000010  000000G        BICB    #10,FLAGS       ;CLEAR NATIVE MODE SELECT
001420  105367  000000G          20$:  DECB    CMBFPN
001424  105267  000000G                INCB    CRBFAV          ;BUFFER FREE FOR INPUT
001430  005720                        TST     (R0)+           ;INCR QUEUE UNLOAD INDEX
001432  020027  000000G                CMP     R0,#CMBMAX      ;AT END OF QUEUE?
001436  002401                        BLT     CMOUT2          ;NO
001440  005000                        CLR     R0              ;YES - RESET TO BEGINNING
001442  110067  000000G          CMOUT2: MOVB    R0,PRINTX       ;UPDATE INDEX
001446  000167  177272          CMOUTX: JMP     CRAGN
```

B	1	RCSRSX	J	5	MRKPNT	E	10	COMTEX	M	14	SCIP
C	1	RCSRSX	K	5	MRKPNT	F	10	RECEIV	N	14	SCIP
D	1	RCSRSX	L	5	MRKPNT	G	10	RECEIV	B	15	SCIP
E	1	RCSRSX	M	5	COMTEX	H	10	RECEIV	C	15	SCIP
F	1	RCSRSX	N	5	INITAL	I	10	RECEIV	D	15	SCIP
G	1	RCSRSX	B	6	COMTEX	J	10	RECEIV	E	15	SCIP
H	1	RCSRSX	C	6	COMTEX	K	10	RECEIV	F	15	SCIP
I	1	RCSRSX	D	6	INITAL	L	10	RECEIV	G	15	WRITE
J	1	RCSRSX	E	6	INITAL	M	10	RECEIV	H	15	WRITE
K	1	RCSRSX	F	6	INITAL	N	10	RECEIV	I	15	WRITE
L	1	RCSRSX	G	6	INITAL	B	11	RECEIV	J	15	WRITE
M	1	RCSRSX	H	6	INITAL	C	11	RECEIV	K	15	WRITE
N	1	RCSRSX	I	6	INITAL	D	11	RECEIV	L	15	WRITE
B	2	RCSRSX	J	6	INITAL	E	11	RECEIV	M	15	WRITE
C	2	RCSRSX	K	6	INITAL	F	11	RECEIV	N	15	WRITE
D	2	RCSRSX	L	6	INITAL	G	11	RECEIV	B	16	COMTEX
E	2	RCSRSX	M	6	INITAL	H	11	RECV	C	16	TRANSM
F	2	RCSRSX	N	6	NXTFIL	I	11	RECV	D	16	COMTEX
G	2	GET	B	7	NXTFIL	J	11	RECV	E	16	COMTEX
H	2	GET	C	7	NXTFIL	K	11	RECV	F	16	TRANSM
I	2	GET	D	7	NXTFIL	L	11	RSXMC	G	16	TRANSM
J	2	GET	E	7	NXTFIL	M	11	RSXMC	H	16	TRANSM
K	2	GET	F	7	NXTFIL	N	11	RSXMC	I	16	TRANSM
L	2	GET	G	7	NXTFIL	B	12	RSXMC	J	16	TRANSM
M	2	GET	H	7	PARSE	C	12	RSXMC	K	16	TRANSM
N	2	GET	I	7	PARSE	D	12	RSXMC	L	16	TRANSM
B	3	GET	J	7	PARSE	E	12	SCP322	M	16	TRANSM
C	3	GET	K	7	PARSE	F	12	SCP322			
D	3	GET	L	7	PARSE	G	12	SCP322			
E	3	GET	M	7	PARSE	H	12	SCP322			
F	3	.MAIN.	N	7	PARSE	I	12	SCP322			
G	3	INTHND	B	8	PARSE	J	12	SCP322			
H	3	INTHND	C	8	QUEHAN	K	12	SCP322			
I	3	INTHND	D	8	QUEHAN	L	12	SCP322			
J	3	INTHND	E	8	QUEHAN	M	12	SCP322			
K	3	INTHND	F	8	QUEHAN	N	12	SCP322			
L	3	INTHND	G	8	QUEHAN	B	13	SCP322			
M	3	INTHND	H	8	QUEHAN	C	13	SCP322			
N	3	INTHND	I	8	RADCX	D	13	SCP322			
B	4	INTHND	J	8	RADCX	E	13	SCP322			
C	4	.MAIN.	K	8	RADCX	F	13	SCP322			
D	4	INTHND	L	8	RADCX	G	13	SCP322			
E	4	INTHND	M	8	RADCX	H	13	SCP322			
F	4	INTHND	N	8	RADCX	I	13	SCP322			
G	4	INTHND	B	9	RADOUT	J	13	SCP322			
H	4	INTHND	C	9	RADOUT	K	13	SCP322			
I	4	INTHND	D	9	RADOUT	L	13	SCP322			
J	4	INTHND	E	9	RADOUT	M	13	SCP322			
K	4	INTHND	F	9	RADOUT	N	13	SCP322			
L	4	INTHND	G	9	RADOUT	B	14	SCP322			
M	4	LOOKUP	H	9	READ	C	14	SCP322			
N	4	LOOKUP	I	9	READ	D	14	SCP322			
B	5	LOOKUP	J	9	READ	E	14	SCP322			
C	5	LOOKUP	K	9	READ	F	14	SCP322			
D	5	LOOKUP	L	9	READ	G	14	SCP322			
E	5	LOOKUP	M	9	READ	H	14	SCP322			
F	5	LOOKUP	N	9	READ	I	14	SCP322			
G	5	MRKPNT	B	10	COMTEX	J	14	SCP322			
H	5	MRKPNT	C	10	RECEIV	K	14	SCP322			
I	5	MRKPNT	D	10	COMTEX	L	14	SCP322			

```
                                                .SBTTL  PARAMETER BLOCKS
                                        ;
001452' 000G                    EOTPAR: .BYTE   BASELN
001453  100                             .BYTE   100
001454' 000000G                         .WORD   EOTSEQ
001456  000001                          .WORD   1
                                        ;
001460  000G                    STXEQ:  .BYTE   BASELN
001461  100                             .BYTE   100
001462  001466'                         .WORD   STEQM
001464  000002                          .WORD   2
                                        ;
001466  0[ ] .0G                STEQM:  .WORD   XSTEQ
```

```
TRANSMIT        MACRO M1104  06-DEC-77 08:29  PAGE 12
CONSTANT DATA AREA

                                                .SBTTL  CONSTANT DATA AREA
                                        ;
001470  077721                  QTRAN:  .RAD50  'TRA'
        000001                          .END
```

TRANSMIT MACRO M1104 06-DEC-77 08:29 PAGE 12-1
SYMBOL TABLE

```
ARND     000104R     002 CMOUT2  001442R     002 F.ERR = ****** GX       NRCVSF= 000000       TAGE5   000050R   003
BASELN= ****** G         CMPAR  = ****** G         GETENT= ****** G       NTRNSF= 000002       TAGNUM= 000005
BINEOT= ****** G         CMPLST= ****** G         GOABRT  000554RG    002 NXTFIL= ****** G     TAG1    000000R   003
BINST    000374R     002 CODE   = ****** G         GOABT1  000526R     002 OFFHAN= ****** G     TAG2    000003R   003
BINTTD= ****** G         COEND   000302R     002 GOG1    000120R     002 PRINTX= ****** G     TAG3    000027R   003
CARC     001054R     002 CRAGN   000744R     002 GOJMP   000056R     002 PROMPT= ****** GX    TAG4    000030R   003
CARCA    001134R     002 CRANW   000772R     002 GOJMP1  000110R     002 QMODE  = ****** G     TAG5    000041R   003
CARCB    001146R     002 CRAOK   001004R     002 GOJMP2  000114R     002 QTRAN   001470RG  002 TDONE  = ****** G
CARCE    001124R     002 CRBASX= ****** G         IE.EOF= ****** GX       RCVBLK= ****** G     TERT    000576R   002
CARC5    001164R     002 CRBFAV= ****** G         INITF  = ****** G       RECSIZ= ****** G     TER1    000620RG  002
CARC6    001212R     002 CRPGN   000740R     002 IOERR  = ****** G       REDS   = ****** G     TER2    000644R   002
CARC7    001224R     002 DFIDN   001054R     002 ITBM   = ****** G       RSX    = 000001       TOT     000666R   002
CHDFIX   000416R     002 ELATCH= ****** G         MASTD   000324R     002 RSX11M= 000001       TRNC    000474R   002
CLOSER= ****** G         EMTPUT= ****** GX       MAXTD   000264R     002 RVIPEN= ****** G     TRNSBL= ****** G
CLOSET= ****** G         ENBPAR= ****** G         MMAS    000366R     002 STEQM   001466RG  002 TURNAR= ****** G
CMBFPN= ****** G         ENDPEN= ****** G         MODE   = ****** G       STOREX= ****** G     TVAB   = ****** G
CMBMAX= ****** G         ENTRY  = ****** G         MRKFIL= ****** G       STXEQ   001460R   002 TXPEN  = ****** G
CMBUFE= ****** G         EOTFLG= ****** G         MSGDON  000000RG    002 SWRBN  = ****** G     TYPMSG= ****** GX
CMBUF1= ****** G         EOTPAR  001452R     002 MSGTA2  000124R     002 SWREN  = ****** G     XEOTS  = ****** G
CMCLST= ****** G         EOTPEN= ****** G         MSGTC   000162R     002 SWTBN  = ****** GX    XFEREN= ****** G
CMFLST= ****** G         EOTSEQ= ****** G         MSGTD   000234R     002 TAGE1   000003R   003 XMTERR= ****** G
CMOUT    001240R     002 FDBIN  = ****** GX       MSTG1   000342R     002 TAGE2   000020R   003 XSTEQ  = ****** G
CMOUTX   001446R     002 FLAGS  = ****** G         NMES    000230R     002 TAGE3   000030R   003 XXX     000570R   002
CMOUT1   001302R     002 FSTGET= ****** G         NOTRNS  000724R     002 TAGE4   000041R   003

. ABS.   000000     000
         000000     001
MAIN     001472     002
MESAG2   000050     003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  1335 WORDS  ( 6 PAGES)
DYNAMIC MEMORY:  2810 WORDS  ( 10 PAGES)
ELAPSED TIME:  00:00:25
 256,020 TRNSMT;27, 256,030 TRNSMT/CRF/-SP=UAPRXM,CTXMAC,TRNSMT
```

TRNSMT CREATED BY MACRO ON 6-DEC-77 AT 08:30 PAGE 1
SYMBOL CROSS REFERENCE CREF

SYMBOL		VALUE		REFERENCES							
ARND		000104	R	5-90	5-93	#5-95					
BASELN	=	******	G	5-68	11-326	11-331					
BINFOT	=	******	G	5-62	6-167	10-287	10-308				
BINST		000374	R	6-162	#6-166						
BINTTO	=	******	G	6-168	10-287	10-309					
CARC		001054	R	#9-246							
CARCA		001134	R	#9-265	9-271						
CARCB		001146	R	9-259	#9-267						
CARCE		001124	R	9-256	#9-263						
CARC5		001164	R	9-253	9-262	#9-270					
CARC6		001212	R	9-269	#9-276						
CARC7		001224	R	9-278	#9-281						
CHDFIX		000416	R	6-161	6-165	#6-169					
CLOSER	=	******	G	5-62	6-208						
CLOSET	=	******	G	5-69	5-135	6-199	9-266				
CMBFPN	=	******	G	5-67	*9-282	10-290	*10-315				
CMBMAX	=	******	G	5-67	9-277	10-318					
CMBUFE	=	******	G	5-66	8-240						
CMBUF1	=	******	G	5-68	8-242						
CMCLST	=	******	G	5-65	*8-235	9-273	*9-275	10-294			
CMFLST	=	******	G	5-66	*9-246	*9-261	*9-268	10-295	10-301		
CMOUT		001240	R	5-129	8-232	#10-288					
CMOUTX		001446	R	10-289	10-291	10-299	#10-322				
CMOUT1		001302	R	#10-296							
CMOUT2		001442	R	10-319	#10-321						
CMPAR	=	******	G	5-62	*5-128	10-288	*10-293	*10-294	*10-295	10-298	
CMPLST	=	******	G	5-65	*8-234	9-272	10-293				
CODE	=	******	G	5-64	6-171	10-311					
COEND		000302	R	#5-134	5-148						
CRAGN		000744	R	6-182	#8-223	10-322					
CRANW		000772	R	8-228	#8-230						
CRAOK		001004	R	#8-233							
CRBASX	=	******	G	5-65	8-230	8-234	*8-238	8-239	*8-242		
CRBFAV	=	******	G	5-64	*6-181	8-223	*9-283	*10-316			
CRPGN		000740	R	6-216	#6-218	8-224	8-226				
DFIDN		001054	R	8-241	#8-243						
ELATCH		******	G	5-60	*5-91	8-225	*9-267				
EMTPUT	=	******	GX	5-123	6-172	6-195	10-298	10-313			
ENBPAR	=	******	G	5-63	*6-169	*6-170	*6-171	6-172	10-287	*10-311	*10-312	10-313
ENDPEN	=	******	G	5-61	*5-103	*9-270					
ENTRY	=	******	G	5-59	5-74						
EOTFLG	=	******	G	5-59	*5-86						
EOTPAR		001452	R	6-195	#11-326						
EOTPEN	=	******	G	5-59	*5-84	*6-211					
EOTSEQ	=	******	G	5-63	*6-167	10-287	*10-303	*10-308	11-328		
FDBIN	=	******	GX	9-252							
FLAGS	=	******	G	5-59	5-77	5-142	*6-153	6-158	*6-166	6-170	*6-173	6-175
				*6-212	*10-310	10-312	*10-314				
FSTGET	=	******	G	5-67	9-260						
F.ERR	=	******	GX	9-252							
GETENT	=	******	G	5-61	5-100	5-125	6-177	6-218			
GOABRT		000554	RG	5-98	5-146	6-185	6-189	#6-192			

TRNSMT CREATED BY MACRO ON 6-DEC-77 AT 08:30
SYMBOL CROSS REFERENCE CREF

SYMBOL		VALUE		REFERENCES							
GOABT1		000526	R	5-97	#6-187						
GOG1		000120	R	5-78	#5-100	5-127					
GOJMP		000056	R	5-87	#5-89						
GOJMP1		000110	R	5-83	#5-97						
GOJMP2		000114	R	5-80	#5-98						
IE.EOF	=	******	GX	9-252							
INITF	=	******	G	5-59	5-75	*6-174	*6-215				
IOERR	=	******	G	5-61	*5-121						
ITBM	=	******	G	5-62	9-274						
MASTD		000324	R	5-76	#5-142						
MAXTD		000264	R	5-122	#5-126						
MMAS		000366	R	6-159	#6-163						
MODE	=	******	G	5-62	5-126	*6-178	6-197	6-206	*6-213		
MRKFIL	=	******	G	5-69	8-229						
MSGDON		000000	RG	5-59	#5-74						
MSGTA2		000124	R	5-85	#5-101						
MSGTC		000162	R	5-102	#5-109						
MSGTD		000234	R	5-110	#5-121						
MSTG1		000342	R	5-143	#6-153						
NMES		000230	R	5-113	5-115	#5-119					
NOTRNS		000724	R	6-196	#6-217						
NRCVSF	=	000000		#1-5							
NTRNSF	=	000002		#1-4							
NXTFIL	=	******	G	5-66	9-255						
OFFHAN	=	******	G	5-60	5-88						
PRINTX	=	******	G	5-67	6-214	10-292	*10-321				
PROMPT	=	******	GX	5-105	5-118	6-190	6-193	6-217			
QMODE	=	******	G	5-64	*6-180						
QTRAN		001470	RG	5-64	6-180	#12-340					
RCVBLK	=	******	G	5-68	6-209						
RECSIZ	=	******	G	5-66	8-236						
REDS	=	******	G	5-65	8-231						
RSX	=	000001		#1-2	4-45	6-157	6-209	9-251	10-305		
RSX11M	=	000001		#1-3							
RT11	=	******		4-48	5-70	6-154	6-210	9-247	10-306		
RVIPEN	=	******	G	5-61	5-92	*5-104	*5-116	*5-136	6-202	8-227	*9-264
STEQM		001466	RG	*6-168	10-287	*10-304	*10-309	11-333	#11-336		
STOREX	=	******	G	5-65	*6-214	8-233	*9-281				
STXEQ		001460	R	5-123	#11-331						
SWRBN	=	******	G	5-63	6-160						
SWREN	=	******	G	5-67	9-258						
SWTBN	=	******	GX	6-163	10-305						
TAGE1		000003	R	5-105	#5-105						
TAGE2		000020	R	5-118	#5-118						
TAGE3		000030	R	6-190	#6-190						
TAGE4		000041	R	6-193	#6-193						
TAGE5		000050	R	6-217	#6-217						
TAGNUM	=	000005		#1-8	*5-105	#5-105	5-105	5-118	#5-118	5-118	6-190
				6-190	6-193	#6-193	6-193	6-217	#6-217	6-217	#6-190
TAG1		000000	R	5-105	5-105	#5-105					
TAG2		000003	R	5-118	5-118	#5-118					
TAG3		000020	R	6-190	6-190	#6-190					

SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE	REFERENCES					
TAG4	000030 R	6-193	6-193	#6-193			
TAG5	000041 R	6-217	6-217	#6-217			
TDONE	= ****** G	5-60	*5-89	5-111	*6-194	6-200	*9-265
TERT	000576 R	5-120	#6-195				
TER1	000620 RG	6-185	#6-199				
TER2	000644 R	6-198	#6-206				
TOT	000666 R	6-201	6-203	6-207	#6-211		
TRNC	000474 R	6-176	#6-178				
TRNSBL	= ****** G	5-64	6-205	9-263			
TURNAR	= ****** G	5-63	5-95	5-137			
TVAB	= ****** G	5-60	*5-79	*5-124			
TXPEN	= ****** G	5-62	*5-94	5-114	*5-134	*6-179	*9-257
TYPMSG	= ****** GX	5-105	5-118	6-190	6-193	6-217	
XEOTS	= ****** G	10-287	10-303				
XFEREN	= ****** G	5-68	6-169				
XMTERR	= ****** G	5-68	*5-81				
XSTEQ	= ****** G	5-68	10-304	11-336			
XXX	000570 R	6-191	#6-194				

MACRO NAME	REFERENCES					
ASRBUF	#3-25					
ERROR	#1-9	5-105	5-118	6-190	6-193	6-217
ERRORK	#1-16	5-105	5-118	6-190	6-193	6-217
ERRORM	#1-32					
ERRORP	#1-47					
FATAL	#1-53					
GETS	#3-43					
LINIT	#3-23					
PUTD	#3-52	5-123	6-195	10-298		
PUTMC	#3-38					
PUTTC	#3-33	#6-172	#10-313			
PUTTM	#3-47					

TABLE OF CONTENTS

2-	44	MACRO CALLS TO RSX
3-	51	MACRO DEFINITIONS
4-	82	Q10 AND FDB BLOCKS
5-	100	START OF CODE
6-	133	ASK FOR VECTOR ADDRESS
7-	157	ASK FOR DEVICE ADDRESS
8-	183	ASK FOR DEVICE PRIORITY
9-	198	ASK FOR LINE BAUD RATE
10-	227	WRITE AND EVENT FLAGS SUBROUTINE
11-	262	TABLES

```
                        .TITLE  CONFIG
                        .IDENT  /00004/

                ;       BATCH CONFIGURATOR FOR THE RSX-11 2780
                ;
                ; COPYRIGHT (C) 1975
                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                ;
                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON
                ; A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH
                ; THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS
                ; SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED
                ; OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON
                ; EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO
                ; THESE LICENSE TERMS.  TITLE TO AND OWNERSHIP OF THE
                ; SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                ;
                ; THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
                ; WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COM-
                ; MITMENT BY DIGITAL EQUIPMENT CORPORATION.
                ;
                ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
                ; OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
                ; DEC.
                ;
                ;       JULY 8,1974
                ;
                ;       BY LOUISE GERHART
                ;
                ; V00-2 9/25/74 RHR
                ; V00-3 6/18/75 RHR
                ;
                ;       DEFINITIONS OF REGISTERS
177776                  PS=177776
000000                  R0=%0
000001                  R1=%1
000002                  R2=%2
000003                  R3=%3
000004                  R4=%4
000005                  R5=%5
000006                  SP=%6
000007                  PC=%7
```

MACRO CALLS TO RSX

```
                        .SBTTL  MACRO CALLS TO RSX
                ;
                        .MCALL  FDOFF$,FCSBT$,FINIT$,GMCR$
                        .MCALL  OPEN$W,PUT$,CLOSE$
                        .MCALL  FDBDF$,FDAT$A,FDRC$A,FDOP$A,FSRSZ$
                        .MCALL  NMBLK$,QIO$,DIR$,WTSE$S,CLEF$S,EXIT$S
```

```
CONFIG  MACRO M1104  08-DEC-77 11:34  PAGE 3
MACRO DEFINITIONS

                                        .SBTTL  MACRO DEFINITIONS
                                ;
                                ;       MESSAGE MACRO
                                ;       TO PRINT MESSAGES
000000                                  TAGNUM=0
                                        .MACRO  MESSAG  TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        MOV     #TAG'N,QDPBM+Q.IOPL
                                        MOV     #TAGE'N-TAG'N,QDPBM+Q.IOPL+2
                                        JSR     PC,EVTFLGS
                                        .PSECT  MESSAG
                                TAG'N:  .ASCII  "TEXT"
                                        .ASCII  /?/
                                TAGE'N:
                                        .PSECT  AAA
                                        .ENDM
                                        .ENDM
                                        .MACRO  ERROR   TEXT
                                        TAGNUM=TAGNUM+1
                                        .IRP    N,<\TAGNUM>
                                        MOV     #TAG'N,QDPBM+Q.IOPL
                                        MOV     #TAGE'N-TAG'N,QDPBM+Q.IOPL+2
                                        JSR     PC,OUPUT
                                        .PSECT  MESSAG
                                TAG'N:  .ASCII  /TEXT/
                                TAGE'N:
                                        .PSECT  AAA
                                        .ENDM
                                        .ENDM
```

CONFIG MACRO M1104 08-DEC-77 11:34 PAGE 4
Q10 AND FDB BLOCKS

```
                                        .SBTTL  Q10 AND FDB BLOCKS
000000                                  .PSECT  AAA
                                ;
000000                          QDPBM:  Q10$    10.WLB,1,1,,,,<,,40>
000030                          QDPBR:  Q10$    10.RLB,1,2,,IOSTB,,<KBIBUF,80.,>
000060                          CMDLN:  GMCR$
        000062'                         KBIBUF=CMDLN+G.MCPB
                                ;
000202                          CMDFDB: FDBDF$
000342                                  FDAT$A  R.VAR,FD.CR
000342                                  FDRC$A
000342                                  FDOP$A  3,,DFNBLK
000342                          DFNBLK: NMBLK$  RJP,CMD,,SY,0
000400                          DFNB2:  NMBLK$  CPUP,MAC,,SY,0
000436                                  FSRSZ$  1
                                ;
000000                          IOSTB:  .BLKW   2       ;I/O STATUS BLOCK
```

```
                                                .SBTTL  START OF CODE
                                                .PSECT  AAA
                                        START:
                                                FINIT$
000436  012700  000202'                          MOV     #CMDFDB,R0      ;GET FDB FOR COMMAND FILE
000442                                          OPEN$W                  ;OPEN FOR WRITE
000446  103451                                  BCS     ERR1
000460  012704  000062'                          MOV     #KBIBUF,R4
                                        ;ASK WHETER SHE HAS A DU OR DP
000462                                  QUS1:   MESSAG  <DP OR DU>
000466  022714  050104                          CMP     #"DP,(R4)       ;IS IT A DP
000506  001015                                  BNE     10$             ;NO
000512  012760  000257'  000026                  MOV     #DPCMD,F.NRBD+2(R0)
000514  012701  000265'                          MOV     #LINE2A,R1      ;REPLACE LINE 2 WITH LINE 2A
000522  012702  000177'                          MOV     #LINE2,R2       ;...
000526  012703  000007                          MOV     #LINE2E,R3      ;LENGTH OF LINE 2A
000532  112122                          5$:     MOVB    (R1)+,(R2)+     ;...
000536  005303                                  DEC     R3
000540  001375                                  BNE     5$              ;GO FOR ALL CHARACTERS
000542  000406                                  BR      20$
000544  022714  052504                  10$:    CMP     #"DU (R4)       ;IS IT A DU
000546  001345                                  BNE     QUS             ;WELL THEN WHAT IS IT
000552  012760  000262'  000026                  MOV     #DUCMD,F.NRBD+2(R0)
000554  012760  000003  000024          20$:    MOV     #3.F.NRBD(R0)   ;3 BYTE RECORD
000562                                          PUT$                    ;WRITE
000570  103413                                  BCS     ERR3            ;WRITE ERROR
000574                                          CLOSE$                  ;CLOSE THE FILE
000576  103012                                  BCC     QUS3            ;CLOSE ERROR
000602
                                        ERR1:   ERROR   <OPEN FAILURE>  ;PRINT ERROR MESSAGE
000604  000167  000516                  ERR3.   JMP     ERR2
000624
```

START OF CODE

```
                                                .SBTTL  ASK FOR VECTOR ADDRESS
000630  012704  000062'          QUS3:   MOV     #KBIBUF,R4
000634  010405                          MOV     R4,R5           ;SAVE ADDRESS OF TTY INPUT
000636  012760  000400' 000046          MOV     #DFNB2,F.DFNB(R0);DEFAULT FILE BLOCK
000644                                  MESSAG  <VECTOR ADDRESS> ;ASK FOR VECTOR ADDRESS
000664  122714  000063                  CMPB    #'3,(R4)        ;IS IT LESS THAN THREE?
000670  003357                          BGT     QUS3            ;YES, ILLEGAL ADDRESS
000672  122724  000067                  CMPB    #'7,(R4)+       ;IS IT A LESS THAN SEVEN?
000676  002754                          BLT     QUS3            ;YES, ILLEGAL ADDRESS
000700  122714  000067                  CMPB    #'7,(R4)        ;IS SECOND DIGIT MORE THAN A 7
000704  002751                          BLT     QUS3            ;YES, ILLEGAL ADDRESS
000706  122724  000060                  CMPB    #'0,(R4)+       ;IS SECOND DIGIT LESS THAN A 0?
000712  003346                          BGT     QUS3            ;YES, REPRINT THE QUESTION
000714  122724  000060                  CMPB    #'0,(R4)+       ;IS THIRD DIGIT A ZERO?
000720  001343                          BNE     QUS3            ;NO, REPRINT THE QUESTION
000722                                  OPEN$W                  ;OPEN THE FILE (CPVP.MAC)
000734  103723                          BCS     ERR1            ;OPEN FAILURE
000736  012701  001456'                 MOV     #FILE,R1        ;POINTER TO LINES AND BYTE COUNTS
000742  012703  000174'                 MOV     #VECT,R3        ;FOR VECTOR ADDRESS
000746  112523                          MOVB    (R5)+,(R3)+     ;MOVE VECTOR ADDRESS FROM KEYBOARD BUF
000750  112523                          MOVB    (R5)+,(R3)+
000752  112523                          MOVB    (R5)+,(R3)+
000754  004767  000346                  JSR     PC,WRITE        ;WRITE VECTOR ADDRESS LINE OF FILE
```

```
CONFIG  MACRO M1104  08-DEC-77 11:34  PAGE 7
ASK FOR DEVICE ADDRESS

     157                                            .SBTTL  ASK FOR DEVICE ADDRESS
     158 000760  012704  000062'          QUS4:    MOV     #KBIBUF,R4      ;ADDRESS OF KEYBOARD BUFFER
     159 000764  010405                            MOV     R4,R5           ;SAVE ADDRESS
     160 000766                                     MESSAG  <DEVICE ADDRESS>
     161 001006  122724  000061                     CMPB    #'1,(R4)+       ;IS FIRST DIGIT A 1?
     162 001012  001362                             BNE     QUS4            ;NO
     163 001014  122724  000066                     CMPB    #'6,(R4)+       ;IS SECOND DIGIT A 6?
     164 001020  001404                             BEQ     10$             ;YES
     165 001022  122764  000067  177777           CMPB    #'7,-1(R4)      ;IS SECOND DIGIT A 7?
     166 001030  001353                             BNE     QUS4            ;NO
     167 001032  012702  000004          10$:      MOV     #4,R2           ;CHECK NEXT FOR DIGITS FO. OCTAL
     168 001036  122714  000067          20$:      CMPB    #'7,(R4)        ;GREATER THAN 7?
     169 001042  002746                             BLT     QUS4            ;YES, REPRINT THE QUESTION
     170 001044  122724  000060                     CMPB    #'0,(R4)+       ;LESS THAN 0?
     171 001050  003343                             BGT     QUS4            ;YES, REPRINT THE QUESTION
     172 001052  005302                             DEC     R2              ;FOUR CHARACTERS CHECKED?
     173 001054  001370                             BNE     20$             ;NO, CHECK SOME MORE
     174 001056  012703  000206'                    MOV     #CSR,R3         ;MOVE DIGITS FROM KEYBOARD BUFFER
     175 001062  112523                             MOVB    (R5)+,(R3)+
     176 001064  112523                             MOVB    (R5)+,(R3)+
     177 001066  112523                             MOVB    (R5)+,(R3)+
     178 001070  112523                             MOVB    (R5)+,(R3)+
     179 001072  112523                             MOVB    (R5)+,(R3)+
     180 001074  112523                             MOVB    (R5)+,(R3)+
     181 001076  004767  000224                     JSR     PC,WRITE        ;WRITE DEVICE ADDRESS LINE OF FILE
```

```
CONFIG  MACRO M1104  08-DEC-77 11:34  PAGE 8
ASK FOR DEVICE PRIORITY

     183                                        .SBTTL  ASK FOR DEVICE PRIORITY
     184 001102  012704  000062'          QUS5: MOV     #KBIBUF,R4        ;ADDRESS OF KEYBOARD BUFFER
     185 001106  010405                         MOV     R4,R5             ;SAVE ADDRESS
     186 001110                                 MESSAG  <DEVICE PRIORITY (DEFAULT = 5)>
     187 001130  005767  000002'                TST     IOSTB+2           ;ANYTHING TYPED?
     188 001134  001413                         BEQ     10$               ;NO, USE DEFAULT
     189 001136  005367  000002'                DEC     IOSTB+2           ;SUBTRACT ONE FROM CHARACTER TYPED
     190 001142  001357                         BNE     QUS5              ;ONLY ONE CHARACTER ALLOWED
     191 001144  121527  000064                 CMPB    (R5),#'4          ;LESS THAN 4?
     192 001150  002754                         BLT     QUS5              ;YES, NO GOOD
     193 001152  121527  000067                 CMPB    (R5),#'7          ;GREATER THAN 7?
     194 001156  003351                         BGT     QUS5              ;YES, NO GOOD
     195 001160  111567  000224'                MOVB    (R5),PRIORT       ;OK, STORE ANSWER
     196 001164  004767  000136          10$:   JSR     PC,WRITE          ;WRITE PRIORITY LINE
```

```
CONFIG  MACRO M1104  08-DEC-77 11:34  PAGE 9
ASK FOR LINE BAUD RATE

    198
    199 001170  012704  000062'          QUS6:   MOV     #KBIBUF,R4      ;ADDRESS OF KEYBOARD INPUT BUFFER
    200 001174  010405                          MOV     R4,R5           ;SAVE ADDRESS
    201 001176                                  MESSAG  <LINE SPEED (BITS/SECOND)>
    202 001216  016702  000002'                 MOV     IOSTB+2,R2      ;GET NUMBER OF CHARACTERS TYPED
    203 001222  001762                          BEQ     QUS6            ;ZERO - ILLEGAL
    204 001224  020227  000005                  CMP     R2,#5           ;GREATER THAN 5?
    205 001230  003357                          BGT     QUS6            ;YES, ILLEGAL
    206 001232  012703  000245'                 MOV     #SPEED+5,R3     ;LAST CHAR + 1 TO STORE ANSWER
    207 001236  060205                          ADD     R2,R5           ;LAST CHAR + 1 TYPED BY USER
    208 001240  114543                  10$:    MOVB    -(R5),-(R3)     ;STORE NEXT CHARACTER
    209 001242  121527  000060                  CMPB    (R5),#'0        ;LESS THAN 0?
    210 001246  002750                          BLT     QUS6            ;YES, ERROR
    211 001250  121527  000071                  CMPB    (R5),#'9        ;GREATER THAN 9?
    212 001254  003345                          BGT     QUS6            ;YES, ERROR
    213 001256  005302                          DEC     R2              ;ONE LESS TO STORE
    214 001260  001367                          BNE     10$             ;MORE TO GO
    215 001262  020327  000240'         20$:    CMP     R3,#SPEED       ;FILLED IN THE AREA AREA?
    216 001266  101403                          BLOS    30$             ;YES
    217 001270  112743  000060                  MOVB    #'0,-(R3)       ;NO, FILL OUT WITH ZEROS
    218 001274  000772                          BR      20$
    219 001276  004767  000024          30$:    JSR     PC,WRITE        ;WRITE SPEED LINE OF FILE
    220 001302  004767  000020                  JSR     PC,WRITE        ;WRITE LAST LINE OF FILE
    221
    222
    223 001306                                  DIR$    #CMDLN
    224 001314                                  CLOSE$
    225 001320                                  EXIT$S
```

```
CONFIG  MACRO M1104  08-DEC-77 11:34  PAGE 10
WRITE AND EVENT FLAGS SUBROUTINE


                                                .SBTTL  WRITE AND EVENT FLAGS SUBROUTINE
                                        ;
                                        ;
                                        ;       ROUTINE TO UPDATE
                                        ;       USER BUFFER POINTER AND USER BUFFER SIZE
                                        ;       AND ISSUE A PUT
001326  012160  000026          WRITE:  MOV     (R1)+,F.NRBD+2(R0)
001332  012160  000024                  MOV     (R1)+,F.NRBD(R0)
001336                                  PUT$
001342  103401                          BCS     ERR2
001344  000207                          RTS     PC
001346                          ERR2:   ERROR   <WRITE ERROR>
                                        ;
                                        ;
                                        ;
                                        ;       ROUTINE TO WAIT FOR EVENT FLAGS
                                        ;       QUES WRITE(THE QUESTION)
                                        ;       WAITS AND THEN QUES READ
                                        ;       AND WAITS FOR A REPLY
                                        ;
001366                          EVTFLGS: DIR$   #QDPBM
001374                                  DIR$    #QDPBR
001402                                  WTSE$S  #2
001414                                  CLEF$S  #2
001426  000207                          RTS     PC

                                        ;
                                        ;
                                        ;       ROUTINE TO QUE WRITE TO
                                        ;       TELTYPE AND THEN EXIT
                                        ;
001430                          OUPUT:  DIR$    #QDPBM
001436                                  WTSE$S  #1
001450                                  EXIT$S
```

```
                                                .SBTTL  TABLES
                                
                                                .PSECT MESSAG
                                                .EVEN
000166      103     120     126 LINE1:  .ASCII  'CPVP=='
000171      120     075     075
000174      060     060     060 VECT:   .ASCII  '000'
000177      104     125     103 LINE2:  .ASCII  'DUCSR=='
000202      123     122     075
000205      075
000206      060     060     060 CSR:    .ASCII  '000000'
000211      060     060     060
000214      104     105     056 LINE3:  .ASCII  'DE.PRI=='
000217      120     122     111
000222      075     075
000224      065     052     064 PRIORT: .ASCII  '5*40'
000227      060
000230      104     056     123 LINE4:  .ASCII  'D.SPED=='
000233      120     105     104
000236      075     075
000240      060     060     060 SPEED:  .ASCII  '00000'
000243      060     060
000245      056     057     061         .ASCII  './100.'
000250      060     060     056
000253      056     105     116 LINE5:  .ASCII  '.END'
000256      104
000257                          LINE6:
000257      100     104     120 DPCMD:  .ASCII  '@DP'
000262      100     104     125 DUCMD:  .ASCII  '@DU'
000265      103     120     122 LINE2A: .ASCII  'CPRSR=='
000270      123     122     075
000273      075
            000007              LINE2E=.-LINE2A
                                .EVEN
001456                          .PSECT  AAA
001456  000166'            FILE:   .WORD   LINE1
001460  000011                      .WORD   LINE2-LINE1
001462  000177'                    .WORD   LINE2
001464  000015                      .WORD   LINE3-LINE2
001466  000214'                    .WORD   LINE3
001470  000014                      .WORD   LINE4-LINE3
001472  000230'                    .WORD   LINE4
001474  000023                      .WORD   LINE5-LINE4
001476  000253'                    .WORD   LINE5
001500  000004                      .WORD   LINE6-LINE5
        000436'                     .END    START
```

```
CONFIG  MACRO M1104  08-DEC-77 11:34  PAGE 11-1
SYMBOL TABLE

CMDFDB  000202R     002 F.DVNM= 000134      F.STBK= 000036      PRIORT  000224R     004 TAGE1   000011R     004
CMDLN   000060R     002 F.EFBK= 000010      F.UNIT= 000136      PS    = 177776            TAGE2   000025R     004
CSR     000206R     004 F.EFN = 000050      F.URBD= 000020      QDPBM   000000R     002 TAGE3   000044R     004
DFNBLK  000342R     002 F.EOBB= 000032      F.VBN = 000064      QDPBR   000030R     002 TAGE4   000063R     004
DFNB2   000400R     002 F.ERR = 000052      F.VBSZ= 000060      QUS1    000466R     002 TAGE5   000121R     004
DPCMD   000257R     004 F.FACC= 000043      G.MCRB= 000002      QUS3    000630R     002 TAGE6   000152R     004
DUCMD   000262R     004 F.FFBY= 000014      IOSTB   000000R     QUS4    000760R     002 TAGE7   000165R     004
ERR1    000604R     002 F.FNAM= 000110      IO.RLB= ****** GX   QUS5    001102R     002 TAGNUM= 000007
ERR2    001346R     002 F.FNB = 000102      IO.WLB= ****** GX   QUS6    001170R     002 TAG1    000000R     004
ERR3    000624R     002 F.FTYP= 000116      KBIBUF= 000062R     002 Q.IOAE= 000012      TAG2    000011R     004
EVTFLG  001366R     002 F.FVER= 000120      LINE1   000166R     004 Q.IOEF= 000006      TAG3    000025R     004
FD.CR = ****** GX       F.HIBK= 000004      LINE2   000177R     004 Q.IOFN= 000002      TAG4    000044R     004
FILE    001456R     002 F.LUN = 000042      LINE2A  000265R     004 Q.IOLU= 000004      TAG5    000063R     004
FO.WRT= ****** GX       F.MBCT= 000054      LINE2E= 000007          Q.IOPL= 000014      TAG6    000121R     004
F.ACTL= 000076          F.MBC1= 000055      LINE3   000214R     004 Q.IOPR= 000007      TAG7    000152R     004
F.ALOC= 000040          F.MBFG= 000056      LINE4   000230R     004 Q.IOSB= 000010      VECT    000174R     004
F.BBFS= 000062          F.NRBD= 000024      LINE5   000253R     004 R.VAR = ****** GX   WRITE   001326R     002
F.BDB = 000070          F.NREC= 000030      LINE6   000257R     004 SPEED   000240R     004 $$$ARG= 000003
F.BGBC= 000057          F.OVBS= 000030      N.DID = 000024          START   000436R     002 $$$OST= 000122
F.BKDN= 000026          F.RACC= 000016      N.DVNM= 000032          S.BFHD= 000020          .CLOSE= ****** G
F.BKDS= 000020          F.RATT= 000001      N.FID = 000000          S.FATT= 000016          .FINIT= ****** G
F.BKEF= 000050          F.RCNM= 000034      N.FNAM= 000006          S.FDB = 000140          .FSRCB= ****** G
F.BKP1= 000051          F.RCTL= 000017      N.FTYP= 000014          S.FNAM= 000006          .OPEN = ****** G
F.BKST= 000024          F.RSIZ= 000002      N.FVER= 000016          S.FNB = 000036          .PUT  = ****** G
F.BKVB= 000064          F.RTYP= 000000      N.NEXT= 000022          S.FNBW= 000017          ...PC1= 000202R     002
F.CHR = 000075          F.SEQN= 000100      N.STAT= 000020          S.FNTY= 000004          ...PC2= 000434R     002
F.CNTG= 000034          F.SPDV= 000072      N.UNIT= 000034          S.FTYP= 000002          ...PC3= 000202R     002
F.DFNB= 000046          F.SPUN= 000074      OUPUT   001430R     002 S.NFEN= 000020          ...TPC= 000020
F.DSPT= 000044

. ABS.  000000      000
        000004      001
AAA     001502      002
$$FSR1  001020      003
MESSAG  000274      004
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  4888 WORDS  ( 20 PAGES)
DYNAMIC MEMORY:  5234 WORDS  ( 20 PAGES)
ELAPSED TIME:  00:00:53
 256,020 CONFIG;27, 256,030 CONFIG/-SP=CONFIG
```